



计 算 机 科 学 丛 书

Springer

# 推荐系统

## 技术、评估及高效算法

弗朗西斯科·里奇 (Francesco Ricci)

利奥·罗卡奇 (Lior Rokach)

[美] 布拉哈·夏皮拉 (Bracha Shapira) 编

保罗 B. 坎特 (Paul B. Kantor)

李艳民 胡聪 吴宾 王雪丽 丁彬钊 译 蒋凡 审核

### Recommender Systems Handbook

FRANCESCO RICCI  
LIOR ROKACH  
BRACHA SHAPIRA  
PAUL B. KANTOR EDITORS

### RECOMMENDER SYSTEMS HANDBOOK

Springer



机械工业出版社  
China Machine Press



# 推荐系统 技术、评估及高效算法

## Recommender Systems Handbook


本书英文名中的“Handbook”一词非常贴切。作为一本推荐系统的手册，它既不是一本泛谈推荐技术的入门书籍，也不是一本纯理论的论文集。本书针对我们在建设推荐系统中会遇到的实际问题，结合理论和应用场景，实践性地进行了深入探讨。推荐系统的初学者可能并不适合直接阅读本书的部分章节，本书更适合作为推荐系统研发人员的参考书，在有一定实践经验后，结合工作中遇到的真实问题来阅读。

—— 廖若芹，调调App创始人，前高德副总裁，前百度技术委员会主席

本书全面介绍推荐系统中的数据处理方法，使读者可能很容易理解推荐系统的原理、算法和实现。无论是对推荐系统的初学者还是业界实践者来说，本书都是一个福音。

—— 唐杰，清华大学副教授，Arnetminer创始人

推荐系统是为用户推荐所需物品的软件工具和技术，对于在线处理信息过载是一个非常有价值的方法，并成为电子商务领域最强大和流行的工具之一。本书融合不同领域专家学者的理论成果和实践经验，从推荐系统相关技术、推荐系统的应用与评估、推荐系统的交互、推荐系统和社区及高级算法5个方面介绍推荐系统的主要概念、理论、方法、趋势、挑战和应用，详细阐释如何支持用户决策、计划和购买过程，帮助你梳理推荐系统的相关知识体系，并理解推荐系统的原理、算法及实现。

 Springer

投稿热线: (010) 88379604  
客服热线: (010) 88378991 88361066  
购书热线: (010) 68326294 88379649 68995259

华章网站: [www.hzbook.com](http://www.hzbook.com)  
网上购书: [www.china-pub.com](http://www.china-pub.com)  
数字阅读: [www.hzmedia.com.cn](http://www.hzmedia.com.cn)



上架指导: 计算机/信息检索

ISBN 978-7-111-50393-4



9 787111 503934 >

定价: 139.00元

计 算 机 科 学 丛 书

# 推荐系统

技术、评估及高效算法

弗朗西斯科·里奇 (Francesco Ricci)

利奥·罗卡奇 (Lior Rokach)

[美] 布拉哈·夏皮拉 (Bracha Shapira) 编

保罗 B. 坎特 (Paul B. Kantor)

李艳民 胡聪 吴宾 王雪丽 丁彬钊 译 蒋凡 审核

## Recommender Systems Handbook

FRANCESCO RICCI  
LIOR ROKACH  
BRACHA SHAPIRA  
PAUL B. KANTOR EDITORS

## RECOMMENDER SYSTEMS HANDBOOK

 Springer



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

推荐系统: 技术、评估及高效算法 / (美) 里奇 (Ricci, F.) 等编; 李艳民等译. —北京: 机械工业出版社, 2015.6

(计算机科学丛书)

书名原文: Recommender Systems Handbook

ISBN 978-7-111-50393-4

I. 推… II. ①里… ②李… III. 计算机网络 IV. TP393

中国版本图书馆 CIP 数据核字 (2015) 第 115324 号

本书版权登记号: 图字: 01-2013-7856

Translation from the English language edition:

*Recommender Systems Handbook* by Francesco Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor.

Copyright © 2011 Springer-Verlag New York, Inc.

Springer is a part of Springer Science+ Business Media.

All Rights Reserved.

本书中文简体字版由 Springer Science+ Business Media 授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

本书汇聚不同领域专家学者的理论成果和实践经验, 全面介绍推荐系统的主要概念、理论、趋势、挑战和应用, 详细阐释如何支持用户决策、计划和购买过程。书中既详细讲解了经典方法, 又介绍了一些新研究成果, 内容涵盖人工智能、人机交互、信息技术、数据挖掘、统计学、自适应用户界面、决策支持系统、市场和客户行为等领域, 无论是从事技术开发, 还是从事产品营销的读者, 都能从中受益。

本书可分成五部分, 共 25 章。第 1 章是概述, 系统介绍推荐系统的概念、功能、应用领域以及当前应用过程中遇到的问题与挑战。第一部分 (第 2~7 章) 介绍当前构建推荐系统最普遍使用的技术, 如协同过滤、基于内容的数据挖掘方法、上下文相关的方法等。第二部分 (第 8~12 章) 概述已用于评价推荐质量的技术和方法, 涉及推荐系统设计与实践方面, 描述设计和实施推荐系统的注意事项, 为选择更合适的算法提供准则, 以及评估用于开发推荐系统的方法、挑战和评测指标。第三部分 (第 13~17 章) 讨论推荐系统如何呈现、浏览、解释和可视化等若干问题, 这一部分讨论的技术使推荐过程更加结构化以及具有可交互性。第四部分 (第 18~21 章) 讨论利用各类用户生成内容 (UGC, 如标签、搜索查询、信任评价等) 产生类型新颖且更可信的推荐结果。第五部分 (第 22~25 章) 讨论推荐系统的高级课题, 如探索用主动学习的原则来指导获取新知识; 防止推荐系统受恶意用户攻击的合适技术; 如何整合多种类型的用户反馈和用户偏好信息来构造更可靠的推荐系统。

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 秦 健

责任校对: 殷 虹

印 刷: 北京瑞德印刷有限公司

版 次: 2015 年 7 月第 1 版第 1 次印刷

开 本: 185mm×260mm 1/16

印 张: 36.25

书 号: ISBN 978-7-111-50393-4

定 价: 139.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

文艺复兴以来，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的优势，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自1998年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson, McGraw-Hill, Elsevier, MIT, John Wiley & Sons, Cengage 等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出 Andrew S. Tanenbaum, Bjarne Stroustrup, Brian W. Kernighan, Dennis Ritchie, Jim Gray, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Abraham Silberschatz, William Stallings, Donald E. Knuth, John L. Hennessy, Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力相助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专门为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

华章网站：[www.hzbook.com](http://www.hzbook.com)

电子邮件：[hzjsj@hzbook.com](mailto:hzjsj@hzbook.com)

联系电话：(010)88379604

联系地址：北京市西城区百万庄南街1号

邮政编码：100037



华章科技图书出版中心



面对互联网上的海量信息，普通用户往往无所适从。为了满足用户的信息需求，计算机科学家发明了两项意义深远的技术：一是搜索技术，它帮助计算机系统被动地应答用户的查询，将与之最相关、最重要的信息返回给用户；二是推荐系统技术，它根据用户的兴趣、行为、情景等信息，把用户最可能感兴趣的内容主动推送给用户。近年来，推荐系统技术得到了长足的发展，不但成为学术研究的热点之一，而且在电子商务、在线广告、社交网络等重要的互联网应用中大显身手。

推荐系统是一个相对而言比较新的技术领域，市面上全面介绍其相关技术的中文书籍并不多见。我和蒋凡在微博上“神交”已久，这次听闻他在翻译整理《推荐系统：技术、评估及高效算法》一书，非常欣慰，也相信他的译本可以为对推荐系统感兴趣的广大读者带来福音。因此，欣然允诺写序文一篇，来“推荐”这本书。

本书是一本结集，由来自不同领域的资深研究人员联合撰写，其内容涵盖了推荐系统的各个方面，既有实战价值，又有理论深度，是一本受众面很广的好书。它的英文版有非常好的销售记录，也被很多学校和公司作为参考书目。将这样一本书翻译成中文有很大的价值，同时也有不小的难度。比如，由于本书篇幅很长，翻译过程需要付出巨大的时间成本；由于本书各章节的多样性，翻译过程需要很宽的知识面；由于推荐系统的技术比较新，某些技术词汇的翻译尚未达成广泛共识；由于本书兼备技术和理论，对译者的算法、数学等方面都提出了很高的要求。翻阅了蒋凡的翻译稿，能够清晰地感受到他在本书的翻译过程中所做的巨大努力。多年来蒋凡对推荐系统技术的深入研究，以及在互联网公司的亲身实践，能为本书的翻译保驾护航。总体而言，本书的翻译稿用词准确、语言流畅，并且在原书交代不够清晰的地方还加入了译者注解，更方便读者阅读和理解。相信蒋凡参与翻译的这本书会对国内推荐系统方面的研究与实践产生积极的推动作用，并且成为该技术领域的重要中文参考文献。

刘铁岩

首席研究员/主任研究员

微软亚洲研究院

2015年6月

随着互联网尤其是移动互联网的兴起，在海量的信息和数据中挖掘有价值的信息呈现给用户，成为电商、社交、新闻、影音等各大主流应用的核心功能。

推荐系统正是这样一项技术，它集数据挖掘、机器学习、用户行为学、人机交互等多个领域的知识为一体，结合日渐普及的大规模并行数据处理框架，为每位用户提供精准的、及时的乃至带有惊喜感的个性化信息服务。

本书以推荐系统领域的欧美知名学者为主要发起人，结合实际应用，为推荐系统的初学者和从业人员提供了一份权威参考资料。

对在校学生及初学者而言，若时间允许，建议比较全面地浏览本书主要章节，边读边思考，重在构建知识的体系、深入思考问题和解决问题的习惯。例如深刻理解协同过滤中 Item-based、User-based(同好)、矩阵分解等算法各自的特性，独立进行代码实现，在开放的、带有语义信息的数据集上，根据多种指标分析推荐的结果并思考算法为何会产生此类结果。通过这个过程，将书本知识转化为自己的体悟。

对推荐系统的从业者而言，本书是一本很好的工作手册，遇到问题时可以随时翻阅，但其作用不仅限于此，同时帮助大家开拓思路、提升系统性解决问题的能力。举例而言，本书第 8 章提到的推荐系统的若干评测指标，如新颖度、多样性、惊喜度、健壮性等，给我们启发良多，在手机淘宝首页逛场景推荐算法和系统中均有应用。对健壮性的考虑，可以预防频频发生的第三方攻击；针对用户惊喜度的设计及多目标优化，可以大大提升产品体验和用户留存率等核心指标。

某些章节介绍的内容较生僻，但在某些垂直行业的推荐系统中可能起着关键作用。如第 21 章中介绍的组推荐算法和策略是多用户参与的旅游活动、家庭娱乐系统的关键；第 20 章谈到的信任推荐问题是社交网络、金融服务领域开发推荐系统的核心要素。

推荐系统是一个年轻的领域，相关技术正在蓬勃发展，学有余力的朋友可以多关注推荐和相关领域顶级会议(SIGKDD、ACM RecSys、WWW、WSDM 等)及国内外业界公司的最新成果。同时，推荐系统的创新应用层出不穷，在移动终端、智能机器人、互联网金融等，只要和用户、消费者交互的地方，都可以看到它的身影。一个更智能、更懂你的互联网离不开智能推荐技术。

本书是一个开始，祝大家早日在推荐的知识海洋中自在遨游，探索未知新世界。

袁全

阿里资深技术专家，淘宝推荐负责人

胡聪(胡户主):这本书着实是一本厚书,在刚开始着手翻译的时候,反复浏览目录里25章的内容,心潮澎湃,激动不已,希望放下手中一切来研习它。在大数据浪潮中,推荐系统作为一个主要分支,逐渐被国内同行所关注,但一直苦于可用资料不多。想必很多刚入此行的同学都在问“有没有全套推荐系统学习资源”,“行业内推荐系统算法有哪些”,等等问题,这本书能够带来一份很好的答案。这本书由概述到细节,由算法理论到行业应用,层层剖析了推荐系统中的技术细节和应用方向,书中知识涵盖较广却又不失细节,相信会成为推荐系统爱好者和从业人员手中必备的一本工具书。

吴宾:Netflix 大赛有力地推进了对推荐系统的广泛而深入的研究,促成新的研究热点。然而,国内当前还缺少比较完整的中文科普图书。因此,本书的出现可有效地填补这一空白,进一步推动推荐系统的研究和应用。综合来看,本书既可以作为推荐系统专业研究人员的入门书籍,也适用于一般应用推荐系统技术的工程人员。对推荐系统的热爱和研究是我有幸参与此书翻译工作的一个重要原因,也因此结识了许多志同道合的小伙伴。我非常感谢他们对我的帮助和指导,与他们的讨论和交流让我受益匪浅。同时,也要感谢艳民和蒋帆老师,他们不惜放弃宝贵的休息时间来审核和校对此书,严谨的工作作风和态度令人敬佩。

丁彬钊(于宁):研究生期间偶然结缘推荐系统,坚信推荐必将成为继搜索之后解决数据过载的重要方法。有机会参与本书的翻译倍感荣幸,虽然期间经历实习、求职、毕业等系列事件,但是在朋友们的支持和帮助下,还是坚持完成了自己负责的那部分翻译工作。通过对此书的翻译,不仅仅提升了自己对推荐系统的理解,也增强了对推荐系统在大数据时代所起作用的信心!特别感谢李艳民对本书翻译提供的各种帮助,以及蒋凡老师细致负责的审核和校对工作!由于翻译水平和时间限制,本书翻译工作难免存在不足之处,欢迎读者朋友批评指正,谢谢!

王雪丽:这是一本“特别”而又“伟大”的书,“特别”在于,它集合了推荐系统所涉及领域的所有知识,并将这些知识有条不紊地展示出来,成为第一本专属于推荐系统领域的书;而“伟大”在于,它凝结了很多人的智慧,比如说每一章后面那长长的参考目录就体现了这点。这本书对于初学者来说是一个很好的起点,不仅可以系统地了解推荐系统的组成部分以及怎样着手设计一个推荐系统,让初学者不会觉得无从下手,另外本书也描述了推荐系统所面临的挑战,也许能在阅读的过程中突发灵感,找到自己的研究方向。另外,非常幸运能够参与本书的翻译,非常感谢一起完成本书翻译的朋友们,也希望读者朋友对我们的工作提出宝贵意见,非常感谢!

李艳民:本书内容和重要程度不言而喻。在长达15个月的翻译过程中共有49位同学参与翻译和审核,期间有晋升为爸爸的、有毕业参加工作的、有结婚步入幸福天堂的,也有开始创业的,在这里由衷感谢他们及其家人能参与和支持我们的翻译。同时特别感谢蒋凡老师能帮我一起审阅。希望本书能给读者带来帮助。读者有疑问或建议可在本书论坛([www.rec-sys.net](http://www.rec-sys.net))讨论。

蒋凡:相信翻译这本书应该是每个推荐系统爱好者的心愿,它的博大精深令人醉心不已,但它的卷帙浩繁也令人望而却步。很幸运能和不计辛劳、只求学问的小伙伴们一起完成这个心愿,翻过高山,收入眼帘的就是一马平川的美景;架起云梯,就能帮助更多的小伙伴攻城拔寨、无往不利。这本书终于面世了,我们用专注和坚韧将它唤醒,是因为我们预感它的魔力能够召唤来更多有志于此的工程师,让技术的力量改变世界。

推荐系统是为用户推荐所需物品的软件工具和技术。提供的推荐旨在通过各种决策过程来支持用户，例如，买什么物品、听什么歌或者读什么新闻。推荐系统对于在线用户处理信息过载是一个非常有价值的方法，并成为电子商务领域最强大和流行的工具。因此，人们提出了各种各样的推荐技术，并在过去的 10 年中将其中很多方法成功地运用在商务领域。

推荐系统的发展需要多学科的支持，涉及来自各个领域的专家知识，如人工智能、人机交互、信息检索、数据挖掘、数据统计、自适应用户界面、决策支持系统、市场营销或消费者行为等。本书旨在基于这种多样性，通过展示推荐系统的主要概念、理论、方法论、趋势、挑战和应用等连贯而又统一的知识体系，帮助读者从差异之中梳理出头绪。这是第一本全面阐述推荐系统的书，其中覆盖了主要技术的多个方面。本书中的丰富信息和实践内容为研究人员、学生和行业中的实践者提供了一个有关推荐系统的全面但简洁方便的参考源。本书不仅详细介绍了经典方法，而且介绍了最近引进的新方法及其扩展。本书由五部分组成：技术、推荐系统的应用和评估、推荐系统的交互、推荐系统和社区及高级算法。第一部分展示了如今构建推荐系统的最流行和最基础的技术，如协同过滤、基于内容的过滤、数据挖掘方法和基于情境感知的方法。第二部分首先介绍用来评估推荐质量的研究技术和方法；其次说明了设计推荐系统的实际方面，如设计和实现的考虑，选择更合适算法的环境指南；再次讨论了可能影响设计的相关方面；最后探讨了应用在已成型系统评估上的方法、挑战和估量。第三部分包括了探讨一系列问题的文章，这些问题包括推荐的展示、浏览、解释和视觉化，以及使得推荐过程更结构化和方便的技术等。

第四部分完全聚焦于一个全新的话题，但该话题却基于过滤推荐的主要思想，例如利用用户产生的各种类型的内容来构建具有新类型并更加可信的推荐系统。

第五部分搜集了一些关于高阶话题的文章，例如利用主动学习技术来引导新知识的学习，构建能够抵挡恶意用户攻击的健壮推荐系统的合适技术，以及结合多种用户反馈和偏好来生成更加可信的推荐系统。

我们要感谢所有为本书做出贡献的作者。感谢所有审阅人员提出的慷慨意见及建议。特别感谢 Susan LagerstromFife 和 Springer 的成员，感谢他们在写这本书过程中的合作。最后我们希望这本手册有助于这一学科的发展，为新手提供一个卓有成效的学习方案，能够激起更多专业人士有兴趣参与本书所讨论的主题，使这个具有挑战性的领域能够硕果累累，长足进展。

Francesco Ricci

Lior Rokach

Bracha Shapira

Paul B. Kantor

2010 年 5 月



出版者的话	
推荐序一	
推荐序二	
译者序	
前言	

<b>第 1 章 概述</b>	<b>1</b>
1.1 简介	1
1.2 推荐系统的功能	3
1.3 数据和知识资源	5
1.4 推荐技术	7
1.5 应用与评价	10
1.6 推荐系统与人机交互	12
1.6.1 信任、解释和说服力	13
1.6.2 会话系统	13
1.6.3 可视化	14
1.7 推荐系统是个交叉学科领域	15
1.8 出现的问题和挑战	16
1.8.1 本书对出现的问题的讨论	16
1.8.2 挑战	18
参考文献	20

## 第一部分 基础技术

<b>第 2 章 推荐系统中的数据挖掘方法</b>	<b>28</b>
2.1 简介	28
2.2 数据预处理	29
2.2.1 相似度度量方法	29
2.2.2 抽样	30
2.2.3 降维	31
2.2.4 去噪	33
2.3 分类	34
2.3.1 最近邻	34
2.3.2 决策树	35
2.3.3 基于规则的分类	36

2.3.4 贝叶斯分类器	36
2.3.5 人工神经网络	38
2.3.6 支持向量机	39
2.3.7 分类器的集成	40
2.3.8 评估分类器	41
2.4 聚类分析	42
2.4.1 k-means	43
2.4.2 改进的 k-means	44
2.5 关联规则挖掘	44
2.6 总结	46
致谢	47
参考文献	47

<b>第 3 章 基于内容的推荐系统：前沿和趋势</b>	<b>51</b>
3.1 简介	51
3.2 基于内容的推荐系统的基础	52
3.2.1 基于内容的推荐系统的高层次结构	52
3.2.2 基于内容过滤的优缺点	54
3.3 基于内容的推荐系统的现状	55
3.3.1 物品表示	56
3.3.2 学习用户特征的方法	62
3.4 趋势和未来研究	65
3.4.1 推荐过程中用户产生内容的作用	65
3.4.2 超越特化：惊喜度	66
3.5 总结	68
参考文献	68

<b>第 4 章 基于近邻推荐方法综述</b>	<b>74</b>
4.1 简介	74
4.1.1 问题公式化定义	75
4.1.2 推荐方法概要	76
4.1.3 基于近邻方法的优势	77
4.1.4 目标和概要	78
4.2 基于近邻推荐	78

4.2.1 基于用户评分 .....	79	5.6 基于邻域的模型和因子分解 模型的比较 .....	127
4.2.2 基于用户分类 .....	80	参考文献 .....	129
4.2.3 回归与分类 .....	80	<b>第6章 开发基于约束的推荐 系统</b> .....	131
4.2.4 基于物品推荐 .....	81	6.1 简介 .....	131
4.2.5 基于用户和基于物品推荐的 对比 .....	81	6.2 推荐知识库的开发 .....	133
4.3 近邻方法的要素 .....	83	6.3 推荐过程中的用户导向 .....	137
4.3.1 评分标准化 .....	83	6.4 计算推荐结果 .....	142
4.3.2 相似度权重计算 .....	85	6.5 项目和案例研究的经验 .....	143
4.3.3 近邻的选择 .....	89	6.6 未来的研究方法 .....	144
4.4 高级进阶技术 .....	90	6.7 总结 .....	147
4.4.1 降维方法 .....	90	参考文献 .....	147
4.4.2 基于图方法 .....	92	<b>第7章 情境感知推荐系统</b> .....	151
4.5 总结 .....	95	7.1 简介 .....	151
参考文献 .....	96	7.2 推荐系统中的情境 .....	152
<b>第5章 协同过滤算法的高级 课题</b> .....	100	7.2.1 什么是情境 .....	152
5.1 简介 .....	100	7.2.2 在推荐系统实现情境信息的 建模 .....	155
5.2 预备知识 .....	101	7.2.3 获取情境信息 .....	158
5.2.1 基准预测 .....	102	7.3 结合情境的推荐系统形式 .....	159
5.2.2 Netflix 数据 .....	103	7.3.1 情境预过滤 .....	161
5.2.3 隐式反馈 .....	103	7.3.2 情境后过滤 .....	163
5.3 因子分解模型 .....	104	7.3.3 情境建模 .....	164
5.3.1 SVD .....	104	7.4 多种方法结合 .....	167
5.3.2 SVD++ .....	105	7.4.1 组合预过滤器案例研究: 算法 .....	168
5.3.3 时间敏感的因子模型 .....	106	7.4.2 组合预过滤器案例研究: 实验结果 .....	168
5.3.4 比较 .....	111	7.5 情境感知推荐系统的其他 问题 .....	170
5.3.5 总结 .....	112	7.6 总结 .....	171
5.4 基于邻域的模型 .....	112	致谢 .....	171
5.4.1 相似度度量 .....	113	参考文献 .....	172
5.4.2 基于相似度的插值 .....	113	<b>第二部分 推荐系统的应用与评估</b>	
5.4.3 联合派生插值权重 .....	115	<b>第8章 推荐系统评估</b> .....	176
5.4.4 总结 .....	117	8.1 简介 .....	176
5.5 增强的基于邻域的模型 .....	117		
5.5.1 全局化的邻域模型 .....	118		
5.5.2 因式分解的邻域模型 .....	122		
5.5.3 基于邻域的模型的动态 时序 .....	126		
5.5.4 总结 .....	127		

8.2 实验设置 .....	177	9.6.1 离线分析 .....	218
8.2.1 离线实验 .....	178	9.6.2 在线分析 .....	220
8.2.2 用户调查 .....	180	9.7 总结 .....	223
8.2.3 在线评估 .....	182	参考文献 .....	223
8.2.4 得出可靠结论 .....	182		
8.3 推荐系统属性 .....	185	<b>第 10 章 走出实验室的推荐系统</b> .....	225
8.3.1 用户偏好 .....	185	10.1 简介 .....	225
8.3.2 预测准确度 .....	186	10.2 设计现实环境中的推荐系统 .....	225
8.3.3 覆盖率 .....	191	10.3 理解推荐系统的环境 .....	226
8.3.4 置信度 .....	192	10.3.1 应用模型 .....	226
8.3.5 信任度 .....	193	10.3.2 用户建模 .....	230
8.3.6 新颖度 .....	194	10.3.3 数据模型 .....	233
8.3.7 惊喜度 .....	195	10.3.4 一个使用环境模型的方法 .....	235
8.3.8 多样性 .....	195	10.4 在迭代设计过程中理解推荐验证步骤 .....	236
8.3.9 效用 .....	196	10.4.1 算法的验证 .....	236
8.3.10 风险 .....	197	10.4.2 推荐结果的验证 .....	237
8.3.11 健壮性 .....	197	10.5 应用实例：一个语义新闻推荐系统 .....	240
8.3.12 隐私 .....	198	10.5.1 背景：MESH 工程 .....	240
8.3.13 适应性 .....	198	10.5.2 MESH 的环境模型 .....	240
8.3.14 可扩展性 .....	199	10.5.3 实践：模型的迭代实例化 .....	243
8.4 总结 .....	199	10.6 总结 .....	244
参考文献 .....	199	参考文献 .....	244
<b>第 9 章 IPTV 服务提供商推荐系统：一个大规模真实产品环境的应用</b> .....	203		
9.1 简介 .....	203	<b>第 11 章 匹配推荐系统的技术与领域</b> .....	247
9.2 IPTV 架构 .....	204	11.1 简介 .....	247
9.3 推荐系统架构 .....	206	11.2 相关工作 .....	247
9.3.1 数据搜集 .....	206	11.3 知识源 .....	248
9.3.2 批处理和实时阶段 .....	207	11.4 领域 .....	250
9.4 推荐算法 .....	208	11.4.1 异构性 .....	250
9.4.1 推荐算法概述 .....	209	11.4.2 风险性 .....	251
9.4.2 基于内容隐语义分析算法 .....	210	11.4.3 变动性 .....	251
9.4.3 基于物品的协同过滤算法 .....	213	11.4.4 交互风格 .....	251
9.4.4 基于降维的协同过滤算法 .....	214	11.4.5 偏好稳定性 .....	251
9.5 推荐服务 .....	215	11.4.6 可理解性 .....	252
9.6 系统评价 .....	216		

11.5 知识源 .....	252	13.4 评价平台中的交互研究 .....	293
11.5.1 社群知识 .....	252	13.4.1 扩展到其他评价平台 .....	294
11.5.2 个人知识 .....	253	13.4.2 用户直接操作与限制用户控制的比较 .....	295
11.5.3 基于内容的知识 .....	253	13.4.3 支持性解释、置信和信任 .....	296
11.6 从领域到技术 .....	254	13.4.4 可视化、自适应性和分区动态性 .....	297
11.6.1 算法 .....	255	13.4.5 关于多文化的适用性的差异 .....	298
11.6.2 抽样推荐领域 .....	256	13.5 评价的评估：资源、方法和标准 .....	298
11.7 总结 .....	257	13.5.1 资源和方法 .....	298
致谢 .....	257	13.5.2 评估标准 .....	299
参考文献 .....	257	13.6 总结与展望 .....	300
<b>第 12 章 用于技术强化学习的推荐系统 .....</b>	<b>261</b>	参考文献 .....	301
12.1 简介 .....	261	<b>第 14 章 构建更值得信任和具有说服力的推荐系统：特性对评估推荐系统的影响 .....</b>	<b>305</b>
12.2 背景 .....	262	14.1 简介 .....	305
12.2.1 TEL 作为上下文 .....	262	14.2 推荐系统作为社交角色 .....	306
12.2.2 TEL 推荐的目标 .....	263	14.3 来源可信度 .....	306
12.3 相关工作 .....	264	14.3.1 可信度 .....	306
12.3.1 自适应教育超媒体 .....	264	14.3.2 专业能力 .....	307
12.3.2 学习网络 .....	265	14.3.3 对来源可信度的影响 .....	307
12.3.3 相同点与不同点 .....	267	14.4 人际交互中信息特性的研究 .....	307
12.4 TEL 推荐系统调查 .....	268	14.4.1 相似度 .....	307
12.5 TEL 推荐系统的评估 .....	271	14.4.2 喜好度 .....	308
12.5.1 对组件的评估 .....	272	14.4.3 权威的象征 .....	308
12.5.2 评估 TEL 推荐系统时需要考虑的问题 .....	273	14.4.4 演讲的风格 .....	308
12.6 总结与展望 .....	274	14.4.5 外在吸引力 .....	308
致谢 .....	274	14.4.6 幽默 .....	309
参考文献 .....	275	14.5 人机交互中的特性 .....	309
<b>第三部分 推荐系统的影响</b>		14.6 用户与推荐系统交互的特性 .....	309
<b>第 13 章 基于评价推荐系统的进展 .....</b>	<b>282</b>	14.6.1 推荐系统类型 .....	310
13.1 简介 .....	282	14.6.2 输入特性 .....	310
13.2 早期：评价系统/已得益处 .....	282	14.6.3 过程特性 .....	311
13.3 评价系统的表述与检索挑战 .....	283		
13.3.1 评价表述的方式 .....	283		
13.3.2 基于评价的推荐系统中的检索挑战 .....	289		



14.6.4 输出特性 .....	311	15.8 总结与展望 .....	336
14.6.5 内嵌的智能体特性 .....	312	参考文献 .....	337
14.7 讨论 .....	312		
14.8 影响 .....	313	<b>第 16 章 基于实例评价研究的产品</b>	
14.9 未来研究方向 .....	314	<b>推荐系统的可用性</b>	
参考文献 .....	314	<b>准则 .....</b>	<b>340</b>
<b>第 15 章 设计和评估推荐系统的</b>		16.1 简介 .....	340
<b>解释 .....</b>	<b>321</b>	16.2 预备知识 .....	341
15.1 简介 .....	321	16.2.1 交互模型 .....	341
15.2 指引 .....	322	16.2.2 基于效用的推荐系统 .....	342
15.3 专家系统的说明 .....	322	16.2.3 准确率、信任度和代价的	
15.4 定义的目标 .....	322	框架 .....	344
15.4.1 系统如何工作: 透明性 ..	324	16.2.4 本章结构 .....	344
15.4.2 允许用户告诉系统它是		16.3 相关工作 .....	345
错误的: 被理解 .....	324	16.3.1 推荐系统分类 .....	345
15.4.3 增加用户对系统上的		16.3.2 基于评分的推荐系统 .....	345
信任: 信任度 .....	325	16.3.3 基于案例的推荐系统 .....	345
15.4.4 说服用户尝试或购买:		16.3.4 基于效用的推荐系统 .....	345
说服力 .....	326	16.3.5 基于评价的推荐系统 .....	346
15.4.5 帮助用户充分地决策:		16.3.6 其他设计指导准则 .....	346
有效性 .....	327	16.4 初始偏好提取 .....	347
15.4.6 帮助用户快速制定决策:		16.5 通过实例激励用户表示	
效率 .....	328	偏好 .....	349
15.4.7 使系统的应用愉悦:		16.5.1 需要多少实例 .....	350
满意度 .....	328	16.5.2 需要哪些实例 .....	350
15.5 评估解释在推荐系统的		16.6 偏好修正 .....	352
作用 .....	329	16.6.1 偏好冲突和部分满足 .....	352
15.5.1 精准度 .....	329	16.6.2 权衡辅助 .....	353
15.5.2 学习效率 .....	329	16.7 展示策略 .....	354
15.5.3 覆盖度 .....	330	16.7.1 一次推荐一项物品 .....	354
15.5.4 接受度 .....	330	16.7.2 推荐 $k$ 项最匹配的物品 ..	355
15.6 用推荐设计展示与互动 .....	330	16.7.3 解释界面 .....	355
15.6.1 展示推荐 .....	330	16.8 准则验证模型 .....	357
15.6.2 与推荐系统交互 .....	331	16.9 总结 .....	359
15.7 解释风格 .....	332	参考文献 .....	359
15.7.1 基于协同风格 .....	333	<b>第 17 章 基于示意图的产品目录</b>	
15.7.2 基于内容风格 .....	334	<b>可视化 .....</b>	<b>363</b>
15.7.3 基于案例风格 .....	334	17.1 简介 .....	363
15.7.4 基于知识/自然语言风格 ..	335	17.2 基于图的可视化方法 .....	364
15.7.5 基于人口统计风格 .....	335	17.2.1 自组织映射 .....	364
		17.2.2 树图 .....	365

17.2.3 多维缩放 .....	366
17.2.4 非线性主成分分析 .....	367
17.3 产品目录图 .....	367
17.3.1 多维缩放 .....	368
17.3.2 非线性主成分分析 .....	369
17.4 通过点击流分析决定属性 权重 .....	370
17.4.1 泊松回归模型 .....	370
17.4.2 处理缺失值 .....	371
17.4.3 使用泊松回归选择权值 .....	371
17.4.4 阶梯式泊松回归模型 .....	371
17.5 图像购物界面 .....	372
17.6 电子商务应用 .....	373
17.6.1 使用属性权值的基于 MDS 的产品目录图 .....	373
17.6.2 基于 NL-PCA 的产品 目录图 .....	375
17.6.3 图像购物界面 .....	377
17.7 总结与展望 .....	379
致谢 .....	380
参考文献 .....	380

## 第四部分 推荐系统与群体

### 第 18 章 个性化 Web 搜索中的 群体、协作与推荐

系统 .....	384
18.1 简介 .....	384
18.2 网络搜索历史简介 .....	385
18.3 网络搜索的未来 .....	387
18.3.1 个性化网络搜索 .....	387
18.3.2 协同信息检索 .....	390
18.3.3 向社交搜索前进 .....	392
18.4 案例研究 1: 基于群体的网络 搜索 .....	392
18.4.1 搜索群体中的重复性和 规律性 .....	392
18.4.2 协同网络搜索系统 .....	393
18.4.3 评估 .....	395
18.4.4 讨论 .....	396
18.5 案例研究 2: 网络搜索共享 .....	396

18.5.1 HeyStaks 系统 .....	397
18.5.2 HeyStaks 推荐引擎 .....	399
18.5.3 评估 .....	400
18.5.4 讨论 .....	402
18.6 总结 .....	402
致谢 .....	403
参考文献 .....	403

### 第 19 章 社会化标签推荐系统 .....

19.1 简介 .....	409
19.2 社会化标签推荐系统 .....	410
19.2.1 大众分类法 .....	410
19.2.2 传统推荐系统范式 .....	411
19.2.3 多模式推荐 .....	412
19.3 现实社会化标签推荐系统 .....	413
19.3.1 有哪些挑战 .....	413
19.3.2 案例 BibSonomy .....	413
19.3.3 标签获取 .....	415
19.4 社会化标签系统的推荐 算法 .....	416
19.4.1 协同过滤 .....	416
19.4.2 基于排序的推荐 .....	418
19.4.3 基于内容的社会化标签 推荐系统 .....	421
19.4.4 评估方案和评估度量 .....	423
19.5 算法比较 .....	424
19.6 总结与展望 .....	426
参考文献 .....	427

### 第 20 章 信任和推荐 .....

20.1 简介 .....	430
20.2 信任的表示与计算 .....	431
20.2.1 信任表示 .....	431
20.2.2 信任计算 .....	433
20.3 信任增强推荐系统 .....	436
20.3.1 动机 .....	436
20.3.2 进展 .....	437
20.3.3 实验比较 .....	441
20.4 进展和开放性挑战 .....	445
20.5 总结 .....	446
参考文献 .....	446

## 第 21 章 组推荐系统 ..... 449

21.1 简介 .....	449
21.2 应用场景和群组推荐系统	
分类 .....	450
21.2.1 交互式电视 .....	450
21.2.2 环绕智能 .....	450
21.2.3 基于场景的推荐系统 .....	451
21.2.4 基于分类的群组推荐 .....	451
21.3 合并策略 .....	452
21.3.1 合并策略概览 .....	452
21.3.2 合并策略在相关工作中的应用 .....	453
21.3.3 哪种策略效果最好 .....	454
21.4 序列顺序的影响 .....	455
21.5 对情感状态建模 .....	456
21.5.1 对个人的满意度进行建模 .....	457
21.5.2 个人满意度对群组的影响 .....	458
21.6 情感状态在合并策略中的使用 .....	459
21.7 对单个用户进行组推荐 .....	460
21.7.1 多准则 .....	460
21.7.2 冷启动问题 .....	461
21.7.3 虚拟组成员 .....	462
21.8 总结与挑战 .....	462
21.8.1 提出的主要问题 .....	463
21.8.2 警告: 组建模型 .....	463
21.8.3 面临的挑战 .....	464
致谢 .....	464
参考文献 .....	465

## 第五部分 高级算法

## 第 22 章 推荐系统中的偏好

### 聚合 ..... 468

22.1 简介 .....	468
22.2 推荐系统中的聚合类型 .....	468
22.2.1 协同过滤中的偏好聚合 .....	470
22.2.2 CB 与 UB 推荐中的特性聚合 .....	470

22.2.3 CB 与 UB 的配置文件构建 .....	470
22.2.4 物品和用户相似度以及邻居的形成 .....	471
22.2.5 基于实例推理的连接词在推荐系统中的应用 .....	472
22.2.6 加权混合系统 .....	472
22.3 聚合函数概论 .....	472
22.3.1 定义和属性 .....	472
22.3.2 聚合成员 .....	475
22.4 聚合函数的构建 .....	479
22.4.1 数据收集和处理 .....	479
22.4.2 期望属性、语义、解释 .....	480
22.4.3 函数表现的复杂度及其理解 .....	481
22.4.4 权重和参数的确定 .....	482
22.5 推荐系统中的复杂聚合过程: 为特定应用定制 .....	482
22.6 总结 .....	485
22.7 进阶阅读 .....	485
致谢 .....	486
参考文献 .....	486

## 第 23 章 推荐系统中的主动

### 学习 ..... 488

23.1 简介 .....	488
23.1.1 推荐系统中主动学习的目标 .....	489
23.1.2 例证 .....	490
23.1.3 主动学习的类型 .....	490
23.2 数据集的属性 .....	491
23.3 主动学习在推荐系统中的应用 .....	492
23.4 主动学习公式 .....	493
23.5 基于不确定性的主动学习 .....	495
23.5.1 输出不确定性 .....	495
23.5.2 决策边界不确定性 .....	496
23.5.3 模型不确定性 .....	497
23.6 基于误差的主动学习 .....	498
23.6.1 基于实例的方法 .....	498
23.6.2 基于模型的方法 .....	500

23.7 基于组合的主动学习 .....	501	第 25 章 具有健壮性的协同推荐 ...	533
23.7.1 基于模型的方法 .....	501	25.1 简介 .....	533
23.7.2 基于候选的方法 .....	502	25.2 问题定义 .....	534
23.8 基于会话的主动学习 .....	504	25.3 攻击分类 .....	536
23.8.1 基于实例的评论 .....	504	25.3.1 基础攻击 .....	536
23.8.2 基于多样性的方法 .....	504	25.3.2 非充分信息攻击 .....	537
23.8.3 基于查询编辑的方法 .....	505	25.3.3 打压攻击模型 .....	537
23.9 计算因素考虑 .....	505	25.3.4 知情攻击模型 .....	538
23.10 总结 .....	505	25.4 检测系统健壮性 .....	539
致谢 .....	506	25.4.1 评估矩阵 .....	539
参考文献 .....	506	25.4.2 推举攻击 .....	540
第 24 章 多准则推荐系统 .....	510	25.4.3 打压攻击 .....	541
24.1 简介 .....	510	25.4.4 知情攻击 .....	542
24.2 推荐作为多准则决策问题 .....	511	25.4.5 攻击效果 .....	543
24.2.1 决策目标 .....	512	25.5 攻击检测 .....	543
24.2.2 准则簇 .....	512	25.5.1 评估矩阵 .....	544
24.2.3 全局偏好模型 .....	513	25.5.2 单用户检测 .....	544
24.2.4 决策支持流程 .....	513	25.5.3 用户组检测 .....	545
24.3 推荐系统的 MCDM 框架:		25.5.4 检测结果 .....	548
经验教训 .....	515	25.6 健壮的推荐算法 .....	548
24.4 多准则评分推荐 .....	517	25.6.1 基于模型的推荐 .....	548
24.4.1 传统的单值评分推荐		25.6.2 健壮的矩阵分解算法 .....	549
问题 .....	517	25.6.3 其他具有健壮性的推荐	
24.4.2 引入多准则评分来扩展		算法 .....	549
传统推荐系统 .....	518	25.6.4 影响力限制器和基于信誉	
24.5 多准则评分推荐算法综述 .....	519	的推荐 .....	550
24.5.1 预测中使用多准则评分 ...	519	25.7 总结 .....	550
24.5.2 推荐中使用多准则评分 ...	524	致谢 .....	551
24.6 讨论及未来工作 .....	526	参考文献 .....	551
24.7 总结 .....	527	本书贡献者名单 .....	554
致谢 .....	528	翻译团队名单 .....	560
参考文献 .....	528		



## 概 述

Francesco Ricci、Lior Rokach 和 Bracha Shapira

**摘要** 推荐系统(Recommender System, RS)是向用户建议有用物品的软件工具和技术。在本章中,我们会简要地介绍推荐系统的基本思想和概念。主要目标是用连贯和结构化的方式描述这本手册的章节内容,以此来帮助读者理解这本手册提供的极其丰富和详细的内容。

## 1.1 简介

推荐系统(RS)是一种软件工具和技术方法,它可以向用户建议有用的物品[60, 85, 25],这种建议适用于多种决策过程,如购买什么物品、听什么音乐、在网上浏览什么新闻等。

“物品”是用来表示系统向用户推荐内容的总称。一个推荐系统通常专注于一个特定类型的物品(如 CD 或新闻),因此它的设计、图形用户界面以及用于生成建议的核心的推荐技术都是为特定类型的物品提供有用和有效的建议而定制的。

推荐系统主要针对的是那些缺乏足够的个人经验和能力的人,他们无法评估潜在的大量可供选择的物品,比如,某个网站提供的商品[85]。一个典型的例子是图书推荐系统,它帮助用户挑选一本书来读。在亚马逊,网站采用个性化推荐系统为每个客户进行推荐[47]。由于推荐通常是个性化的,不同的用户或用户组接收的建议是不同的。当然也存在非个性化推荐。它们都是非常简单的,通常出现在报纸或杂志上。典型的例子包括书籍和 CD 等的 top 10 推荐(最热销的前 10 名)。虽然在某些情况下它们可能是有用和有效的,但这些类型的非个性化推荐通常不是推荐系统研究要解决的问题。

个性化推荐最简单的形式是提供一个排好序的物品列表。通过这个排序列表,推荐系统试图根据用户的偏好和其他约束条件来预测最合适的产品或服务。为了完成这样的计算任务,推荐系统收集用户的喜好,这种喜好是显式的,如为产品打分,或通过解释用户的行为做出推断。例如,推荐系统可能会把访问某个特定商品详情页的行为作为该用户喜爱这个主页上的商品的隐式信号。

推荐系统的发展源于一个很简单的现象:人们在做日常工作和日常决策时总是依赖于其他人提供的建议[60, 70]。例如,要选择一本书的时候,通常依靠朋友的推荐;雇主依靠推荐信做招聘的决定;当选择观看的影片时,人们倾向于阅读并且依赖影评家写在报纸

---

Francesco Ricci, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy e-mail: fricci@unibz.it

Lior Rokach, Department of Information Systems Engineering, Ben-Gurion University of the Negev, Israel  
e-mail: liorrk@bgu.ac.il

Bracha Shapira, Department of Information Systems Engineering, Ben-Gurion University of the Negev, Israel  
e-mail: bshapira@bgu.ac.il

翻译:燕山大学-王孝先 审核:王二朋,承皓,严强

上的影评。

为了模拟这种行为,第一个推荐系统通过算法将社区用户的建议推荐给一个活跃用户,也就是在寻找推荐的用户。推荐的物品是这些相似用户(那些品味相似的用户)喜欢的。这种方法称为协同过滤,它的理论依据是,如果这个活跃的用户以前与一些用户有相似爱好,那么来自这些相似用户的其他推荐应该是相关的,这些推荐也是此活跃用户感兴趣的。

随着电子商务网站的发展,迫切需要出现一种能过滤所有可用的替代品的推荐系统。让用户从这样纷繁复杂的物品(产品和服务)中作出最恰当的选择是非常困难的。

互联网信息的爆炸式增长和种类的纷繁复杂以及新兴电子商务服务(购买产品、产品比较、拍卖等)的出现经常压得用户喘不过气来,导致用户无所适从。这种选择多样性不但没有产生经济效益,反而降低了用户满意度。大家都明白可选择是好的,但是太多的选择就不是很好了。实际上,随着选择对自由、自治和自我决定带来的影响变得过度,我们就会因为选择而觉得自由是一件痛苦的事情[96]。

近年来,推荐系统被证明是一种解决信息过载问题的有效工具。从根本上来讲,推荐系统是通过为用户指引该用户不熟悉的新物品来解决信息过载现象的,这些新物品或许与该用户当前的需求相关。对于用户每一个清晰表达的请求,根据不同的推荐方法和用户所处的环境和需求,推荐系统利用存储在自定义数据库的关于用户、可用物品以及先前交易的数据和各种类型的其他知识产生推荐内容。然后用户可以浏览推荐的内容。用户可能接受也可能不接受推荐,也可能马上或者过一段时间提供隐式或者显式的反馈。所有这些用户的行为和反馈可以存储在推荐数据库,并且可用于在下次用户和系统相互作用时产生新的推荐。

如上所述,相比于其他经典的信息系统的工具和技术(如数据库或搜索引擎),推荐系统的研究是相对较新的。在20世纪90年代中期,推荐系统成为一个独立的研究领域[35, 60, 70, 7]。近年来,下面的事实说明人们对推荐系统的兴趣大大增加:

- 对于一些有很高评价的网站,如 Amazon.com、YouTube、Netflix、Yahoo!、Tripadvisor、Last.fm 和 IMDb,推荐系统扮演了重要的角色。此外,许多媒体公司正在开发和部署推荐系统作为它们提供给用户的服务的一部分。例如,Netflix(一家提供在线电影租赁服务的网站)向第一个能够成功地显著提高推荐系统性能的团队奖励了100万美元[54]。
- 本领域有专门的正式会议和相关的专题研讨会。在这里专指 ACM 推荐系统会议(RecSys),该会议成立于2007年,现在是新的推荐技术的研究和应用的顶级年度盛会。此外,像数据库、信息系统和自适应系统领域等更传统的大会也经常包括以推荐系统为主题的会议。在这些会议中,值得一提的是 ACM SIGIR(Special Interest Group on Information Retrieval)、UMAP(User Modeling, Adaptation and Personalization)和 ACM SIGMOD(Special Interest Group on Management Of Data)。
- 世界各地的高等教育机构在本科生和研究生的课程中已经提供专门研究推荐系统的课程;在计算机科学会议中关于推荐系统的专题报告也引起了人们的极大关注;最近一本介绍推荐系统技术的书也已经出版[48]。
- 在学术期刊中,已经有一些特刊专题涵盖推荐系统领域的研究和发展。期刊中包括推荐系统专刊的有:《AI Communications》(2008)、《IEEE Intelligent Systems》(2007)、《International Journal of Electronic Commerce》(2006)、《International Journal of Computer Science and Applications》(2006)、《ACM Transactions on

Computer-Human Interaction》(2005)和《ACM Transactions on Information Systems》(2004)。

本章简要介绍了推荐系统的基本思想和概念。主要目标并不是对推荐系统给出包罗万象的综合介绍和概览,而是用一个连贯和结构化的方式描述这本手册的章节内容,以此帮助读者理解这本手册提供的极其丰富和详细的内容。

本书分成五部分:推荐相关技术、推荐系统的应用和评价、推荐系统的交互、推荐系统和社区、主流算法。

- 第一部分介绍当前构建推荐系统最普遍使用的技术,例如,协同过滤、基于内容的数据挖掘的方法和上下文相关的方法。
- 第二部分概述了已用于评价推荐质量的技术和方法。这一部分还涉及推荐系统设计与实践方面;描述设计和实施推荐系统的注意事项;为选择更合适的算法提供准则。这部分还考虑到可能影响推荐系统设计(域、设备、用户等)的方面。最后是评估开发的推荐系统所用的方法、挑战 and 评测指标。
- 第三部分包括一些涉及推荐系统如何呈现、浏览、解释和可视化等若干问题的论文。这里讨论的技术使推荐过程更加结构化以及具有可交互性。
- 第四部分讨论的是一个相当新颖的话题,利用各类用户生成内容(UGC,如标签、搜索查询、信任评价等)来产生类型新颖且更可信的推荐结果。尽管这个话题相对新颖,但它实质上根植于基于协同推荐核心理念。
- 第五部分展示了关于几个高级课题的论文,例如,探索如何用主动学习的原则来指导新知识的获取;防止推荐系统受恶意用户攻击的合适技术;如何整合多种类型的用户反馈以及用户偏好信息来构造更可靠的推荐系统。

## 1.2 推荐系统的功能

在前面的内容中,将推荐系统定义为软件工具和技术,这些工具和技术用于为用户提供物品建议信息,而这些推荐信息用户或许会利用。现在我们要完善该定义,以此来表明推荐系统可以起到一系列可能的作用。首先,我们必须区别推荐系统在服务提供商和用户这两方面分别起到的作用。例如,旅游中间商(如 Expedia.com)或目的地管理机构(如 Visitfinland.com)通常会采用旅游推荐系统来增加营业额,比如,销售更多的房间,或者增加目的地的游客数量[86]。然而,用户访问这两个系统的主要动机是在去某个目的地旅游时,找到合适的酒店和有趣的事件或景点。

事实上,服务提供商采用这种技术的原因是多样的:

- **增加物品销售数量。**这可能是商用推荐系统最重要的作用,即与没有使用任何一种推荐系统相比,能卖出一些额外的物品。能做到这一点是因为推荐物品可能适合了用户的需求和愿望。或许用户在尝试过几个推荐后,会认识到这一点<sup>①</sup>。非商业应用也有类似的目标,虽然用户在选择物品的时候无须任何费用。例如,一个基于内容的新闻推荐系统的目的是提高网站上新闻类物品的阅读量。通常,从服务提供商的角度来看,引入推荐系统的主要目的是提高转化率,即接受推荐并消费物品的用户数量比上仅浏览了这些信息的普通访客数量。

① 说服用户接受推荐的问题,会在我们解释预测用户对一物品的兴趣与用户挑选推荐物品的可能性之间的差异时再次进行讨论。

- **出售更多种类的物品。**一个推荐系统的另一个主要功能是让用户能够选择物品。如果没有这样精准的推荐系统，用户可能很难找到自己要选择的物品。例如，对于像 Netflix 这样的电影推荐系统，服务提供者感兴趣的是要能租出去清单里的所有 DVD，而不仅仅是最热门的。如果没有推荐系统，这就很困难了。因为服务提供商没法保证推销的电影就一定能适合特定用户的口味。相反地，推荐系统就能向合适的用户建议或推销不那么热门的电影。
- **增加用户满意度。**一个设计良好的推荐系统还可以提高网站或者应用程序的用户体验。用户会觉得推荐结果既有趣又相关，而且如果人机交互设计得合理，他还会乐于使用这个系统。准确有效的推荐结果，加上可用的用户接口，这些措施将会增加用户对系统的主观评价。反过来这将增加系统的使用率和推荐结果被接受的可能性。
- **增加用户忠诚度。**用户应该忠于一个网站，这样当用户访问该网站时，该网站能够识别出老客户并把他作为一个有价值的访问者。这是一个推荐系统的固有特性，因为许多推荐系统是利用用户之前与网站的交互信息来产生推荐结果的，这些交互信息包括该用户对物品的评分记录等。因此，用户与网站的交互时间越久，用户模型就变得越精确，即用户偏好的系统表示越准确；与此同时，更多的推荐能有效地匹配用户的喜好。
- **更好地了解用户需求。**推荐系统另一个重要的功能是用来描述用户的喜好（这一功能能够在许多其他应用中使用），这些喜好是收集到的显式反馈或者由系统预测到的（隐性反馈）。然后，为了其他目的，如提高物品的库存管理或生产管理，服务提供商可能重新使用这方面的知识。例如，在旅游领域，目的地管理机构可以通过分析由推荐系统收集的数据（用户的交易），决定为一个特定区域的新客户部门做宣传或为特定类型的促销信息发布广告。

我们在上述中提到了电子服务提供商引入推荐系统的几个重要动机。但是如果推荐系统能够有效地支持用户的任务或目标，他们或许也会考虑使用推荐系统。因此推荐系统必须平衡这两类参与者的需求，并为两者都提供有价值的服务。

Herlocker 等[25]的一篇论文已经成为这一领域经典的参考文献，文中定义了推荐系统能够实现的 11 个常见的功能。有些可能通常被视为与推荐系统相关的主要或核心任务，即可能会对用户有用的物品推荐。其他的可能更多地被视为以“机会主义”的方式开发的。事实上，这种任务的分化与使用搜索引擎所遇到的事情很相似，搜索引擎的主要功能是找到用户需求信息的相关文件，但它也可以用于检查一个网页的重要性（在查询结果列表中该页的位置）或发现一个词在文档中的不同用法。

- **发现一些好的物品：**根据用户对物品的喜欢程度（例如，1 到 5 星的评分）的预测结果，我们可以用列表榜单的形式推荐给用户一些物品。对商业系统来说，这是推荐系统的主要任务（见第 9 章）。有些系统不显示具体的预测评分。
- **发现所有好的物品：**推荐所有能满足用户需求的物品。在这种情况下，仅找到一些合适的物品是不够的。当相关物品的数量很少或当推荐系统扮演关键角色时，推荐系统要格外精确。例如，推荐系统在医疗或金融领域的应用。在这些情况下，除了仔细评估所有可能性带来的好处，用户还可以从推荐系统的物品排序或者推荐系统额外产生的解释（推荐理由）中受益。
- **产品注解：**在给定的上下文语境中（如一个物品推荐列表），我们根据用户的长期

偏好来确定这些物品的重要性。例如，一个电视推荐系统会标识在电子节目菜单中出现的哪些电视节目是值得观看的(第18章提供了这个任务的一个有趣实例)。

- **推荐系列产品**：这个思路是将物品的序列作为一个令用户满意的整体进行推荐，而不是集中产生单一的推荐。典型的例子包括推荐一个电视节目系列；推荐一本数据挖掘的书之后，还推荐一本关于推荐系统的书；或者是一个音乐曲目的总集[99]，[39]。
- **搭配推荐**：提供一组完美搭配的物品。例如，一个旅行计划可能由某个特定区域的不同景点、目的地和住宿服务组成。从用户的角度来看，用户可以考虑这些不同的替代方案，并选择一个唯一的旅游目的地[87]。
- **闲逛**：在这项任务中，用户只是简单地浏览目录而并不带有强烈的购买意图。推荐系统的任务是帮助用户浏览一些其在特定浏览时期可能会感兴趣的物品。自适应超媒体技术[23]已经支持该服务。
- **发现可信的推荐系统**：有些用户不信任推荐系统，于是他们抱着试试看的态度去尝试推荐系统的推荐结果。因此，除了那些仅仅想要获得推荐的用户，一些系统还提供特定的功能让用户测试推荐系统的行为。
- **完善用户资料**：这涉及用户向推荐系统提供自己偏好信息的能力。为了提供个性化推荐，这是一个基本任务，也是绝对必要的。如果系统不了解活跃用户(即寻求推荐帮助用户)的明确信息，那系统只能提供给该活跃用户与一般用户一样的推荐。
- **自我表达**：有些用户可能一点也不在意给出的推荐结果。相反，对他们重要的是，系统允许他们展示他们的评分，并表达他们的评价和观念。这种行为产生的用户满意度仍会有助于保持用户和应用程序的紧密关系(正如我们前面讨论服务提供商的动机中提到的那样)。
- **帮助他人**：有些用户乐于贡献信息，如他们对物品的评价(评分)，因为他们相信社区能从他们的贡献中获得益处。这可能是将这类信息输入推荐系统的主要动机，即使他们不经常使用这个推荐系统。例如，汽车推荐系统中，一个已经买了新车的用户将评价信息输入推荐系统时，他更多考虑的是这些信息对于其他用户有用，而不是为了下次再买一辆车。
- **影响他人**：在基于 Web 的推荐系统中，某些用户的主要目标是对其他购买特定产品的用户产生明显的影响。实际情况是，有一些恶意用户可能利用系统仅仅用来促进或抑制某些物品的销售(见第25章)。

正如这些观点所表明的那样，信息系统中的推荐系统的作用是有很大区别的。这种多样性需要利用一系列不同的知识源和技术。在接下来的两节中，我们将讨论一个推荐系统的数据和推荐系统常用的核心技术。

### 1.3 数据和知识资源

推荐系统是信息处理系统，为了实现推荐，该系统会积极收集各种数据。数据主要是关于推荐的物品和收到这些推荐结果的用户，但是由于推荐系统获得的数据和知识来源可能区别很大，它们最终是否可以被利用取决于推荐技术(见1.4节)。这种推荐技术将在本手册的不同章节中有更加清晰的阐述(详见第11章)。

通常，有的推荐技术用的知识(领域)较少，如只用到用户评分或者对物品的评价等一



些简单且基础的数据(第4、5章)。其他的技术则依赖于更多的知识,如利用用户或物品的本体性描述(第3章),或者约束性条件(第6章),或者用户的社交关系和行为活动(第19章)。在任何情况下,推荐系统使用的数据一般指的是三种对象:物品、用户和事务,也就是用户和物品的关系。

**物品。**物品是被推荐的对象集。物品具有复杂性和有价值或效用的特点。如果物品对于用户是有用的,那么物品的作用就是积极的;否则如果物品对于用户不适合,那物品的作用就是消极的,从而导致用户在选择时做出错误决定。我们注意到,当一个用户需要获得一个物品时,他总会付出代价,其中包括搜索物品的认知代价和最终为物品支付的费用。

例如,一个新闻推荐系统的设计师必须考虑新闻物品的复杂性,即新闻的结构、文本表述和任何新闻物品的时变重要性。但是推荐系统的设计者必须意识到,即使用户没有花钱阅读新闻,他们还是为搜索并阅读这些新闻物品付出了认知上的代价。如果选择的物品和用户相关,这个代价就被用户得到有用信息的收益覆盖了,可一旦不相关,推荐物品对用户的净价值就是负面的在其他领域,如汽车或者金融投资,当选择最恰当的推荐方法时,物品真正的货币成本成为一个考虑的重要因素。

复杂度低且价值小的物品是:新闻、网页、书籍、光盘、电影。复杂度高且价值大的物品是:数码相机、手机、个人电脑等。我们通常认为最复杂的物品是保险政策、金融投资、旅游、工作[72]。

根据其核心技术,推荐系统可以使用物品集的一系列属性和特征。例如,在电影推荐系统中,我们可以使用电影种类(如喜剧、悲剧等)信息、导演信息和演员信息来描述电影,并用于弄清楚物品的效用是如何依赖其特征的。物品集可以用各种信息和表述方法来表示,如可以简约到单一的ID码,或者丰富到为一组属性,甚至是该领域本体表示中的一个概念(第3章)。

**用户。**正如前面提到的那样,推荐系统的用户可能有非常不同的目的和特点。为了使推荐结果和人机交互个性化,推荐系统使用用户的一系列信息。这种信息可以用不同的方式组织,而且同样的是,选择哪种信息建模取决于推荐技术。

例如,在协同过滤中,所有用户被建模为一个简单的列表,该列表包含每个用户对若干个物品的评分记录。在基于人口统计学的推荐系统中,会用到如年龄、性别、职业和受教育程度等社会统计学特征。用户数据用来构成用户模型[21, 32]。用户模型简明扼要地描述了用户的特征,即对用户偏好和需求进行编码。不同的建模方法已经在推荐系统中得到应用,在一定意义上,推荐系统可以被看作一个构建并使用用户模型来产生推荐的工具[19, 20]。如果没有一个实用的用户模型,个性化推荐是不可能的(除非推荐系统本身就是像top10那样是非个性化的),因此用户模型起到非常重要的作用。例如,再次考虑协同过滤方法,要么用户被直接描述为他对物品的评价记录,要么系统根据该用户的评分记录构建一个因子向量,不同用户之间的区别是这些因子权重在他们模型中的差别(第4、5章)。

用户也可以通过他们的行为模式数据来描述,如网站浏览模式(在基于Web的推荐系统中)[107]或旅游搜索模式(在旅游推荐系统中)[60]。此外,用户数据包括用户间的关系,如用户间的信任级别(第20章)。推荐系统能利用这些信息给用户推荐物品集,而这些物品集也是相似用户或可信任用户所喜欢的。

**事务(transaction)。**我们一般将一个事务看作用户和推荐系统进行交互的一条记录。



人机交互过程中产生的这种类似日志的数据存储着重要的信息,并且这些数据对系统中推荐生成算法是有用的。比如,事务日志可能会涉及用户选择物品,以及特定推荐所处上下文描述信息(如用户的目的和查询词)。如果可能,事务也会包括用户提供的显式反馈,如对选择物品的评分。

实际上,评分是推荐系统收集交易数据最流行的方式。这些评分可能是用显式或者隐式的方式收集的。收集显式评分时,用户需要在某个评级尺度内给出自己对物品的看法。根据[93],评级可以采用各种方式:

- 数字评分,就像 Amazon.com 的书籍推荐系统中的 1 到 5 星的评价。
- 序数评价,例如,“强烈同意,同意,一般,不同意,强烈反对”,用户在其中选择最能代表自己观点的术语(一般是通过问卷调查)。
- 二元制评价,用户仅仅被要求确定一个物品的好或者不好。
- 一元制评价,用来表示用户已经看到或者买了一个物品,或由此对物品进行明确的评价。在这种情况下,评分值的缺失意味着关联用户和物品的信息是未知的(也许用户在其他地方买了这个物品)。

另一种评价的形式就是关联用户和物品的标签。例如,在 Movielens 推荐系统(<http://movielens.umn.edu>)中,标签表示 Movielens 用户对电影的感觉,例如,“太长”或“表演不错”。第 19 章侧重讲解这类交互行为。

从事务中隐式收集用户评级,系统的目标是根据用户的行为推断用户的意图。例如,如果用户在亚马逊网站输入“瑜伽”进行搜索,那么她将得到关于书的一个很长的列表。作为回报,用户为了获得额外信息会单击列表中的书。从这点上看,系统可以推断用户对那本书有一定的兴趣了。

在会话(conversational)系统中,即支持交互过程的系统中,交易模型更加精确。在这些系统中,用户请求和系统行为交替出现(见第 13 章)。更准确地说,用户请求一个推荐,系统就产生一个推荐列表。但是系统仍然需要额外的用户偏好信息,以期产生更好的结果。在这个交易模型中,系统收集各种请求—响应信息,并且最终通过观察推荐过程的结果来修改系统的交互策略[60]。

## 1.4 推荐技术

为了实现其核心功能,即识别对用户有用的物品,推荐系统必须预测出有推荐价值的物品。为了实现上述目的,系统必须能预测一些物品的效用性(the utility of some of them),或者至少对物品的效用性作对比,然后根据比较决定该推荐的物品。虽然推荐算法中对预测这一步描述不是很明确,但是我们仍然可以使用统一的模型来描述推荐系统的一般作用。我们的目的是提供给读者一个统一的思路,而不是所有不同推荐方法的总和,这些方法会在本书中逐步讲解。

为了说明推荐系统中的预测这一步,我们考虑一个简单的非个性化的推荐算法,该算法仅推荐最流行的歌曲。这个方法的理论依据就是在不知道用户偏好这样更精确的信息的情况下,有理由认为一首流行度高的歌曲会被很多用户喜欢,所以很可能也被一个普通用户喜欢,因此选择一首流行度高的歌曲的效果肯定会比随机选取的歌曲好。因此,这些流行度高的音乐的有效性对于普通用户理所当然就要高。

文献[3]中提到把核心推荐计算过程当作预测某件物品对某个用户的效用的观点。他们把用户  $u$  对物品  $i$  的可用度建模为实数值函数  $R(u, i)$ ,在协同过滤中通常指的是用户

对物品的评分。然后协同过滤推荐系统的主要任务是通过用户与物品对来预测  $R$  的值,即计算  $\hat{R}(u, i)$ , 这里把  $\hat{R}$  作为估计值, 并由此得到真实函数  $R$  的值。接下来, 在物品集上算出活跃用户  $u$  的预测值也就是  $\hat{R}(u, i_1), \dots, \hat{R}(u, i_N)$  之后, 系统将会选择最大效用的物品  $i_{j_1}, \dots, i_{j_K} (K \leq N)$  作为推荐结果。 $K$  通常是个小的数字, 要远小于物品集的基数, 或者说是用户效用预测计算所依赖的物品的基数, 即推荐系统“过滤”出推荐给用户的物品。

正如前面所说, 一些推荐系统在做出推荐之前不是计算全部效用, 但是它们可以应用很多启发式的方法猜测一个物品对用户是否有用。典型的案例是专家系统。这些效用预测是通过特殊算法(如下)计算的, 并使用了各种各样的关于知识、用户、物品和效用函数本身的知识(见 1.3 节)[25]。例如, 系统假定效用函数是布尔型的, 只需要确定一个物品是否对用户有用。因此, 对于一个请求推荐的用户, 如果系统能够获得该用户的一些知识(有可能得不到), 如物品信息以及曾经接收到推荐结果的其他用户的知识, 系统就能使用恰当的算法利用这些信息产生各种效用预测并且据此产生推荐[25]。

我们有必要注意, 有些时候物品对用户的效用是依赖于其他可变因素的, 我们把这些可变因素统称为“语境”[1]。例如, 物品对用户的效用会被用户的领域信息所影响(例如, 使用数码相机的高手与新手), 或者随着推荐请求发生的时间而变化。或者用户可能对离他当前位置近的物品(如一家饭店)更感兴趣。因此, 推荐结果必须与这些特殊的附加信息相适应。但是这样做的结果是, 准确地计算出正确的推荐越来越困难。

本书介绍了许多不同种类的推荐系统, 这些推荐系统根据使用的信息和用户领域知识的不同而变化, 但正如一开始所说那样还是主要随着推荐算法而变化, 即如何预测推荐内容的效用。其他的不同是推荐系统最终如何组合推荐结果并如何把它展示给用户用以响应用户的请求。这些方面在导言的后面也会讨论。

为了提供不同类型推荐系统的综述, 我们引用[25]提出的分类法, 此分类方法是区分推荐系统的一种经典的方法。文献[25]为六种不同的推荐方法做了划分:

**基于内容(content-based):** 系统为用户推荐与他们过去的兴趣类似的物品。物品间的相似性是基于被比较的物品的特征来计算的。例如, 如果某个用户对一部喜剧电影有了正面的评价, 那么系统就能学会从喜剧类型中(为该用户)推荐其他电影。第 3 章给出了基于内容的推荐系统的综述, 梳理出了在涉及系统设计和实现时广泛性和多样性方面的关系, 除此之外, 还介绍了基于内容推荐的基本概念、专业术语、高层次的体系结构以及主要的优点和缺点。接下来对在几个应用领域内使用的最先进系统做了概述。报告还包括了在如何表示物品和用户信息方面的经典方法和最新技术的全面描述。最后讨论了推荐系统的趋势和引领下一代推荐系统的未来研究。

**协同过滤(collaborative filtering):** 这种方法是找到与用户有相同品味的用户, 然后将相似用户过去喜欢的物品推荐给用户, [93]是对这种方法有最简单和最原始的实现。两用户间的相似偏好是通过计算用户历史评分记录相似度得到的。这也是[94]将协同过滤比作“人对人的相互关系”的原因。协同过滤被认为是推荐系统最流行和最广泛实现的技术。

第 4 章介绍了基于邻域的协同过滤方法。这种方法关注物品之间的关系或者用户之间的关系。基于物品的方法依据同一用户对与某个物品的相似物品集的评分, 对用户对该物品的偏好建模。基于邻域的方法享有相当的知名度, 因为它们简单、高效以及具有产生精确计算和个性化推荐的能力。作者将会说明实现基于邻居推荐系统所必需的结论, 并且提供了得到这些结论所需的信息。

最后,第4章还涉及了数据稀疏和覆盖率有限的问题,而这些问题经常出现在大型电子商务推荐系统中。本章提出了这些问题的一些解决方法。

第5章介绍了构建协同过滤推荐系统的最新进展。作者特别介绍了隐语义模型(LFM),例如,矩阵因子分解(如奇异值分解、SVD)。这些方法把用户集和物品集映射到同一个隐语义空间。这样就可以通过因子同时表示产品和用户,利用隐语义空间解释根据用户反馈自动推断出的评分。作者解释了SVD如何处理这些数据的额外特征,包括隐式反馈和时间信息。还描述了克服基于邻域的技术存在的缺点的技术,为此他们建议使用包含了全局优化技术的更加严格的数学公式。利用这些技术我们就能够放宽邻域大小的限制,同时又能把隐性反馈和动态时序信息增加到模型中。预测结果的准确性与矩阵因子分解模型的准确性很接近,同时又提供了一些实用的优势。

**基于人口统计学的(demographic):**这种类型的推荐系统推荐物品时是基于人口统计信息的。我们假设不同的人群信息应该产生不同的推荐。许多网站采用基于人口统计学的简单而有效的个性化解方案。例如,根据用户的语言或者国籍,划分到特定的网站。或者根据用户的年龄定制推荐。虽然这种方法在营销的文献中相当流行,但是推荐系统方面对此一直很少研究[59]。

**基于知识(knowledge-based):**基于知识的系统根据特定的领域知识推荐物品,这些知识是关于如何确定物品的哪些特征能够满足用户需要和偏好,以及最终如何确定物品对用户有用。著名的基于知识的推荐系统是基于案例的[22, 87]。在这些系统中,相似函数用来估算用户需求(问题描述)与推荐(解决问题)的匹配度。这里的相似性得分,可以理解为用户推荐的可用性。

基于约束的系统是另一种基于知识的推荐系统(第6章)。从使用的知识(knowledge)方面来说,这两个系统是相似的:收集用户需求;当找不到解决方法的时候系统自动提供不合理需求的修改方案;并且能对推荐做出解释。这两者主要的不同在于解决方案的计算方法。基于案例的推荐是基于相似的方法,然而,基于约束的推荐系统主要是利用预定义的知识库,这些库包括了如何把消费者需求和物品特征相关联的明确规则。

在刚开始部署的时候,基于知识的系统往往比其他方法工作得更好。但是如果它们没有配备自动学习组件,那么其他利用用户日志或人机交互(如在协同过滤中的交互)信息这样浅显方法的系统就可能超越它。

**基于社区(community-based):**这种推荐方法依赖用户朋友的偏好。这种技术有个业界流行的表述“告诉我你的朋友是谁,我将知道你是谁”[8, 14]。有证据表明,人们往往会更依赖朋友的建议而不是陌生人的建议[103]。这个现象的出现,加上日益普及的社交网络,于是基于社区的推荐系统就应运而生,并且得到人们越来越多的关注。这种系统通常叫作社会化推荐系统[34]。这种推荐系统获取用户的社会关系和用户朋友的偏好等信息并以此进行建模。推荐结果基于用户朋友提供的评分。实际上,这类推荐系统是随着社交网络产生的,可以简单而又全面地采集与用户社交关系相关的数据。

在这个方面的研究还处于起始阶段,并且关于系统性能的定论还很混乱。例如,[34, 64]对社交网络推荐系统做了全面的报告。报告显示,一般情况下,这类系统的准确性没有传统的协同过滤方法高,除非是用户对一个特殊物品的评价变化特别大(即有争议的物品)或者冷启动等特殊的情况。在这种情况下,用户不能提供足够多的评分去计算与其他用户的相似度。其他研究人员表明,在某些情况下,利用社交网络的数据产生的推荐结果比使用用户资料相似度数据产生的推荐结果更加准确[37],并且把社交网络数据增加到传

统的协同过滤算法中能够提高推荐结果的准确性[36]。第20章概述了该领域的研究成果，并对当前结论进行了分析。

**混合推荐系统(hybrid recommender system)**：这类推荐系统综合了上面提到的技术。混合推荐就是综合A和B方法，利用A的优势弥补B的不足。例如，协同过滤方法遭遇的新物品冷启动问题，也就是不能推荐尚未被评分的物品。而基于内容的方法没有这个限制，因为新物品的预测是基于物品描述(特征)，而这些特征都是容易获得的。给出两个(或多个)基本的推荐技术，采用一些方法将这些技术综合起来产生一个混合的系统(文献[25]有详细描述)。

我们说过，当用户需要推荐的时候，基于上下文的方法能使系统推荐更加个性化。例如，考虑时间上下文这个因素时，冬天和夏天的假期推荐应该有很大的区别。或者如饭店推荐系统，工作日和同事一起吃饭的推荐应该和周六晚上与朋友吃饭的推荐有很大区别。

第7章介绍了上下文的一般概念，还介绍了推荐系统是如何对它建模的。此外，作者还提供了案例用于研究如何使几种上下文敏感的推荐技术整合为一个单一算法的可能性。

在这里讨论了把上下文信息整合到推荐过程的三个不同的算法范式：化简(预过滤)、上下文的后过滤以及上下文建模。在基于化简(预过滤)方法中，仅当信息与当前使用的上下文相匹配时(如在同一上下文的物品评分)才能用于计算推荐结果。在上下文后处理方法中，推荐算法会忽略上下文信息。该方法的推荐结果会被过滤或调整，以便只包含与目标上下文相关的推荐结果。上下文建模方法是三种方法中最复杂的方法。在该方法中，上下文信息在预测模型中被显式地使用。

## 1.5 应用与评价

推荐系统的研究着重放在实践和商业应用上。因为除了理论方面的贡献，这方面的研究一般旨在切实促进商业推荐系统的发展。因此，推荐系统的研究包括实现这些系统的实践方面。这些方面与推荐系统生命周期的不同阶段都相关，即系统设计、实现以及系统运行过程中的维护和改善。

系统设计阶段所需考虑的影响因素或许会影响算法的选择。第一个要考虑的因素——应用的领域是算法选择的主要影响因素。[72]提供了推荐系统的分类，并且对特定应用领域的推荐系统应用做了分类。基于这些特定的应用领域，我们为最普遍的推荐系统应用做了更一般的领域分类：

- 娱乐：电影、音乐和IPTV的推荐。
- 内容：个性化新闻报纸、文件推荐、网页推荐、电子学习程序和电子邮件过滤。
- 电子商务：为消费者推荐要购买的产品，如书籍、照相机、电脑等。
- 服务：旅游服务推荐、专家咨询推荐、租房推荐或者中介服务。

随着推荐系统的流行，一些新应用上的潜在优势激发了人们的兴趣。例如，可以像在www.tweeter.com网站那样推荐一个朋友列表或者一个推特信息列表。因此，诸如上面的列表并不能覆盖推荐技术所涉及的所有应用领域，这些列表只是给出了不同类型的应用领域的初始描述。

特定应用领域的推荐系统开发商应该了解该领域的特殊方面、具体的要求、应用程序面临的挑战和局限性。只有在分析这些因素之后，才能选择最佳推荐算法，并且设计更高效的人机交互。

本书第11章提供了推荐技术与应用领域相匹配的指南。Burke 和 Ramezani 写的章节中为推荐系统提供了一个新的分类。与以往推荐系统分类方法(如[25, 94, 3, 7])不同, Burke 和 Ramezani 采用以人工智能为核心的方法, 而且重点关注不同推荐方法所需要的知识源, 并把与这些推荐方法相关联的约束条件作为选择算法的入门指南。该章讨论了针对不同问题选择的不同推荐技术的适用性, 并提出了选择这些技术的决策准则。

该章的目的很明确, 就是为系统实现者“推荐”合适的推荐技术。作者描述了不同领域的推荐系统中可用的知识库以及确定一种推荐技术时所不可获取的知识库。这就意味着设计推荐系统首先要注重的是分析知识库的来源, 然后再据此选择算法。

第12章记述了另一个需要根据领域来调整推荐方法的示例, 这个示例关于推荐系统的技术增强型学习(TEL)。TEL 基本覆盖了支持各种形式的教学活动的技术, 其目的是设计、发展和测试新方法与技术来增强个人和组织双方的学习实践能力。TEL 的主要收益来源于将推荐系统技术整合到个性化学习过程中, 并根据用户先前的知识、能力和偏好逐步调整。该章阐述了能满足 TEL 的特定需求的推荐系统; TEL 设置阶段用户需要做的任务; 其他推荐系统的用户任务和这些任务有什么区别。例如, 对 TEL 来说, 用户的特殊任务是“发现新资源”, 也就是尝试仅仅推荐新的或者新奇的物品。或者, 我们考虑另一个例子, 即“发现新路径”, 也就是通过学习知识库推荐可选的路径。该章提出的过滤方法的分析对 TEL 很有用, 有关现有 TEL 系统的调查显示, 推荐技术已经被部署到系统中。

第10章讨论了推荐系统发展的实践方面, 目的在于为推荐的设计、实现和个性化系统的评测提供指导。设计推荐系统的时候, 除了考虑预测算法, 也要考虑其他因素。第10章列了许多因素: 目标用户的种类和他们的上下文、需要用的设备、推荐系统在程序中的作用、推荐的目标, 还有之前提到的可用的数据。

作者建议为此环境搭建一个三维模型, 三个维度分别是: 系统用户、数据特征和程序的综合应用。推荐系统的设计要基于这个模型。作者阐述了其指导方案以及建立在他们自己开发的一个新闻推荐系统上的模型。

有关推荐系统实际部署方面的另一个重要的问题就是对推荐系统的测评。出于各种各样的目的, 推荐系统生命周期的不同阶段都要进行测评[25, 1]。在设计时, 需要用测评去判定推荐算法的选择。在设计阶段, 测评过程是离线完成的, 并且推荐算法需要拿来和用户交互作比较。离线测评包括在同一个用户互动(如评分)数据集上运行多个算法并且比较各自的性能。如果能够获得合适的数据, 这种测评通常都是在一些公开的基准数据上进行的, 否则, 就要在自己收集的数据集上进行。为了确保结果的可靠性, 离线实验的设计需要与已知的实验设计方法保持一致[11]。

系统上线运行以后依然需要评测。当解决一些核心推荐问题(如预测用户评分)时, 算法可能需要非常精确, 但是有些情况下系统可能不会被用户接受, 例如, 当系统的表现和用户期望的不一致时。实施在线评测并且分析系统日志以增强系统性能就显得很有必要了。另外, 大多数算法都包含参数, 如阈值权重、近邻的数量等, 都需要不断地调整和校准。

另一种测评方法是当在线测评不可行或太冒险的时候, 可以进行一个集中的用户研究(a focused user study)。在这类测评中, 需要实施一种可控制的实验, 即一小群用户被要求执行随着系统变化的不同的任务。然后才有可能分析用户喜好以及分发调查问卷以使用户报告各自的用户体验。在这样的实验中, 我们一般能够收集到关于系统的数量大、质量高的用户反馈信息。



第12章也讲了 TEL 系统环境下的测评方法,对测评方法和工具进行了详细的分析。这些方法和工具可以用来在一组特定的标准下评估 TEL 推荐技术,这组标准是与每个选择的部分(如用户模型、领域模型、推荐策略和算法)相对应的。

第8章详细阐述了用于评估推荐系统的三种实验,并阐述了它们各自的优缺点,最后为测评选择方法制定了指导原则。通常文献中有关评估的讨论都会提到算法预测的精准度及相关度量方法,本章则独树一帜,专门讨论定向特性的评估。除了精准度,还给出了一大组特性。每个特性都有相应的实验验证和度量方法。这些特性包括覆盖率、冷启动、信心值、可信度、新颖性、风险度和惊喜度。

当讨论推荐系统的实践方面时,我们最好分析真实实现的系统。我们的想法是在理论上验证直观的假设,目的是确定它们在实践中是否成立。但是我们必须面对的事实是,商业推荐系统的所有者通常不愿意公开他们真实的做法并且甚至连这样合作的机会几乎也没有。

第9章介绍了一个这样的时机并阐述了一个真实推荐系统的操作,由此说明了适用于推荐系统实践阶段和评估过程的实践层面。Fastweb 是欧洲最大的 IPTV 服务商,本章着重介绍了如何把推荐系统整合到 Fastweb 的生产环境中。本章描述了需求和考虑因素,包括规模和准确度,而这些正是选择推荐算法的主导因素。除此之外,本章还描述了系统中用到的离线和在线评测方法,并且说明了系统应该如何进行相应的调整。

## 1.6 推荐系统与人机交互

正如前面阐明的那样,研究人员主要关注一系列技术解决方案的设计,以及利用各种知识库更好地预测目标用户喜欢什么和喜欢的程度。这种研究活动背后的基本假设是仅能展示正确的推荐(也就是最好的选项)就足够了。换句话说,系统给用户推荐,如果是正确的,用户应该明确地接受推荐。很明显,这极度简化了推荐问题,其实把推荐结果传递给用户并不简单。

实际上,因为用户没有足够的知识库做明确的决定,所以才需要推荐。让用户评价系统提供推荐并不是件容易的事情。因此,很多研究人员试图了解让指定用户接受推荐结果的因素[105, 30, 24, 97, 33]。

其中[105]第一个指出推荐系统的有效性依赖于很多因素,而不仅仅是预测算法的质量。实际上,推荐系统必须说服用户去尝试(或听、购买、阅读、看)所推荐的物品。当然,这取决于被选择物品的个体特征,也取决于推荐算法。当物品被推荐、比较、解释(解释为什么被推荐)的时候,该过程也取决于系统支持的特定的人机交互。[105]发现,从用户角度来看,一个有效的推荐系统必须能激发用户对系统的信任;它必须有一个显而易见的系统逻辑(个人理解为系统不能有逻辑错误);它必须指引用户面向新的、用户不熟悉的物品;它必须提供推荐物品的详细信息,包括图片和社区评分;最终,[105]提供改进推荐的方法。

[105]和其他相似方向的研究者并没有忽略推荐算法的重要性,但是他们声称推荐算法的有效性不能仅根据预测结果的准确性(即用标准且流行的 IR 指标,如 MAE、准确度或 NDCG)来衡量(见 8.5 节,第 9 章)。他们认为涉及用户接受推荐系统以及其推荐结果的其他角度也应该被测量。这些想法在[33]中也被明确地提出并讨论。在这项工作中,作者提出评价推荐系统要以用户为中心,包括推荐列表的相似度,推荐系统有意外新发现的能力,以及推荐系统中用户需求和期望的重要性。

讨论过[105]提出的观点之后,我们介绍由人机交互研究引出的几个重要观点,我们将在本书中进一步研究人机交互。

### 1.6.1 信任、解释和说服力

首先讨论信任。本书讨论了关于信任的两种不同概念:对推荐系统其他用户的信任和对推荐系统本身的信任。

第20章集中讨论了第一种观念,并分析了一类叫作“社会化推荐系统”的推荐系统。这些系统试图从用户资料和用户关系中推导出更有用的推荐,这些资料和关系在目前几乎随处可见,如在 Facebook、LinkedIn 和 MySpace 等社交网站中。由于基于信任的推荐系统主要利用它们在社交网站发现的信任关系来构造新的推荐算法(如[34]),所以它们在核心评分预测问题上只能使用信任关系。这类系统的主要优势是用户可以知道推荐系统的本质是什么,如这些推荐结果是如何确定的,这样用户便会更加信任这些推荐结果。换句话说,我们可以利用用户之间的相互信任来提高用户对推荐系统的信任度。

对推荐系统产生的推荐结果的信任在第15章讨论。该章主要介绍了解释推荐的作用,还介绍了信任是推荐解释在推荐系统扮演的七大作用之一。这七大作用是:透明度,即说明系统是如何工作的;可反馈性,即允许用户告诉系统有错误[50];信任,即增加用户对系统的信心;有效性,即帮助用户做出好的决定;说服力,即说服用户去尝试或购买;高效性,即帮助用户快速抉择;满意度,即增加用户舒适度或乐趣。

该章也阐述了一系列构造解释的方法。如协同过滤方法,其解释的形式是“跟你相似的用户喜欢这个物品”。在基于内容的风格解释中,系统会根据最能代表物品的特征属性把物品推荐给用户。例如,电影推荐系统中,一种推荐形式有可能是“这部电影被推荐是因为你可能喜欢 Bruce Willis 这个明星”,或者“物品 X 被推荐是因为你对物品 Y 和 Z 评价很高,而 Y、Z 和 X 有共同的属性 A 和 B”。用基于案例的解释,系统会引用与推荐物品相似的物品,例如,“这个物品被推荐是因为你说自己拥有物品 X”或者“这些物品的推荐是基于你最近浏览的物品”。最后,用基于知识库的风格解释,系统会解释推荐物品和另一个物品的差异,并且会说推荐物品是如何符合用户目标的:“这个房间面朝大海,并且比之前推荐的房间大,这将更加浪漫,如你所愿。”

回到信任的话题上来,我们把它视为实现推荐系统主要目标,即说服用户接受推荐结果并且尝试推荐的物品。这个问题最终与整个推荐系统的说服力有关,即推荐系统的各个因素,包括推荐什么,如何推荐,在人机交互过程中实际是如何运作的。这个话题将在第14章有所讨论。在这里,作者强调说推荐系统应该值得信赖并且经常被人们想到。当人们想到使用推荐系统时,不仅仅是因为他们知道它能产生推荐结果,而且是把它当作一个建议给予者。实际上,关于说服力的文献资料表明人们容易接受来自可靠消息的推荐,所以我们得到结论:推荐系统的可靠性对于增加用户接受推荐结果的可能性是至关重要的。因此,作者讨论了推荐系统的可信度是如何提高的,并且提供了有关信任度研究的概要。

### 1.6.2 会话系统

推荐系统算法的另一个严格限制是这些算法被设计为一次性收集输入数据,而且一旦返回推荐结果便结束整个过程。在很多情况下,这种模型会失效,因为用户还不能完全意识到自己的喜好,除非他们与系统进行一定程度交互并且大致了解选择的范围。或者在确

信一些可能适合自己的推荐之前会浏览更多的可选择的选项。这也可能意味着系统一开始的推荐就是错误的,但用户可能愿意提供额外的信息用于修复这些问题,最终得到一些好的推荐。

这些问题已经得到了致力于“会话式推荐系统”[27, 110, 67, 60]的研究人员的重视并解决。会话式推荐系统为预测评分或排序使用了各种技术。然而,这些技术都在试图提供交互过程,该过程中,用户和系统能彼此查询或提供信息。最具决定性的问题是如何设计这个会话(也就是会话策略),以及如何设计用户和系统在交互的各个阶段必须执行的动作。提供的会话必须有效,即当用户终止会话时应该得到解决方案(如预订机票),而且速度要快(会话步骤较少)。在本手册中,有两章处理这一重要话题。

第13章提供了基于评价的推荐系统的研究综述。给定一个关于用户偏好信息的初始集(例如,一些物品特征的优选值),基于评价的界面或者会话模型能够展示给用户推荐的物品以及支持用户表述“评价”,例如,“给我展示更多像A的物品,但是比A更便宜”。

基于评价的系统已经在一些领域中得到了很大的关注,在这些领域中,我们需要一个更加复杂且具有交互性的决策/推荐支持系统,如旅游领域[88, 32, 100]或者计算机系统领域[82, 83]。基于评价的系统最初的设计是用于有效地解决用户偏好提取问题的,但是现在已经对其他目的或应用变得非常重要,例如,小组推荐、初始混合推荐、用户自适应交互接口、解释推荐、移动推荐。

另一种与会话系统有关的方法是基于偏好的[67]。基于偏好和基于交互的方法相似,因为他们都是在之前就给用户一些推荐,这些推荐也许不会是最好的,但是总有一部分的物品能引起用户的兴趣。这种额外的信息能帮助提高系统表现用户偏好(用户模型)的能力,从而能使系统产生新的且更好的推荐。

第16章阐述了这些新颖的方法和系统,集中讲了这种基于偏好推荐系统的用户,即系统交互的三个方面:初始偏好提取;偏好修正;推出推荐结果。本章的内容可以作为来源于一些系统的可用性指南的集合,这些指南可被广泛和大规模应用。而且,为了选择指导准则,作者不仅关注准确性这一个指标,还考虑到这样的一个事实:人类的认知资源有限,因此如果需要我们付出过多的努力,将不大可能会得到高准确度的结果。他们选择的方法能产生准确度很高的推荐结果,同时要求的努力程度是我们愿意接受的。

之前提到的方法(基于交互的和基于偏好的)已经主要应用于基于案例的推理系统[22],这类系统的检索部分是依赖相似性度量的。在这种情况下,由于在一定程度上,产品总是跟一个可探测(查询)结果类似,所以一条查询可以检索出目录下的所有产品,并对这些产品进行排序。如果查询语言支持其他限制条件(例如,等同限制或排序限制),查询返回的结果可能不会令人满意[47, 71, 31]。针对这种情况,研究人员提出了许多技术,这些技术通过放宽约束的最小数量来修复这种查询问题,其目的是使返回的查询结果令人满意。这个课题也在专讲基于约束(第6章)的推荐系统中有所讨论。

### 1.6.3 可视化

目前,我们重点强调了人机交互问题,而这些问题已经在推荐系统的研究中得以解决,本手册也对此进行了讨论。总之,我们认为系统如何展示并且可视化计算得到的推荐结果是影响用户接受推荐结果和该推荐系统的关键因素。

推送(将推荐推到用户那里)和解释技术是不可分的:一个好的推送技术也要有解释推荐的能力,也要诱导用户提出进一步请求,包括对解释推荐的请求。目前已有的这些技术

有一个共同点,就是推荐结果按照物品列表的方式显示。推荐列表长度是可变的,但是核心推荐算法的输出通常是个排序列表,该列表在最终展示推荐结果时会用到。

本书单设一章阐明一种有别于传统模式的推送方法。在第17章,作者注意到在排序列表这种可视化方法中丢失了大部分信息。这里的原因是:对于两个产品,即使它们都与用户请求或者用户模型相匹配,但如果是基于完全不同的产品特征集,也是有区别的。如果用一个二维的基于图的可视化推荐,还可能保留部分信息。在图中,我们可以在一个受限的范围内为各个相似推荐安排位置。该章提出了建立二维图推荐的两种方法,并且讨论了各自的优缺点。

## 1.7 推荐系统是个交叉学科领域

设计和开发推荐系统是多个学科共同努力的结果,这得益于多个计算机科学领域的成果,特别是机器学习、数据挖掘、信息检索和人机交互。这在本书章节中和上述讨论可以明显地看出来。在此,简要地梳理一下这些关系。

机器学习和数据挖掘,是人工智能的子领域,它们是让计算机学习示例、数据或过去的经验从而使其执行某些任务时性能最佳[109]。例如,购买“Da Vinci Code”的顾客也买了“The Five People You Meet in Heaven”,数据挖掘就可以利用这类交易数据进行学习。因此,可以用这种相互关联信息构建推荐系统。

为了预测用户对物品的评价,或者为了学习如何修正推荐列表排序,许多推荐系统以利用各种机器学习和数据挖掘算法为中心。本书第2章对上下文推荐系统预处理方法使用到的数据挖掘技术做了总述,例如,抽样或降维;分类技术,如贝叶斯网络和支持向量机的方法;聚类技术,如k-means算法;关联规则。

其他例证推荐系统和数据挖掘关系的章节是:第12章讨论了利用主动学习获取信息;第5章建立预测评分模型的最优化技术;第7章介绍利用上下文数据提供了各类预测评分模型;第24章介绍数据挖掘技术,利用多准则的物品评价标准,更好地预测用户综合评价;第25章重点讲了数据挖掘的推荐系统攻击检测方法,建立健全性算法的解决方案;第4章阐述目前基于学习的协同过滤系统的很多实例;第19章阐述了复合矩阵或超边(用户、资源、标签)的超图的数据挖掘解决方法;第20章介绍数据挖掘的信任网络解决方法。

信息检索(IR)的目的是帮助用户存储和检索各种不同形式的内容,如文本、图像、视频[63]。利用信息检索工具,用户可以很快找到自己需要的有关的信息和全面的信息。虽然IR并不始于Web,但由于Web搜索引擎的发展,万维网在开创这一新观念方面扮演了重要的角色。

信息检索和推荐系统都面临着过滤和排序问题。IR一般重点发展全局检索技术,通常忽略个体需要和用户偏好。而[25]声称,推荐系统还没有明显地与信息检索分离。推荐系统和信息检索或搜索引擎最核心的区别是“考虑个体”与“用户感兴趣和对用户有用”,而推荐系统正尝试在这方面取得成功。

目前,现代的Web搜索引擎已经依赖推荐技术来解决Web搜索的挑战,实现更先进的搜索功能。例如,搜索引擎给用户查询推荐相似查询。各种搜索引擎也试图通过产生用户查询结果实现个性化,返回的结果不仅根据用户查询条件还要参考用户上下文(例如,他的位置)和他的检索历史。

第18章讨论了IR的研究目标,涉及从推荐系统角度实现Web检索个性化。作者阐



述了如何应用推荐系统技术解决搜索引擎面临的挑战。该章的重点是改进搜索引擎的两个有希望实现的想法：个性化和协同。该章描述了利用用户偏好和上下文信息影响搜索结果的一系列不同的方法，去实现个性化 Web 搜索。另外，该章讨论了协同信息检索方面目前要做的工作，协同信息检索尝试利用潜在的合作关系(朋友间、同事间或者有相似需求的用户间)实现一系列的信息检索任务。这种新型的研究叫作社交检索，得益于社交网络中的特性，检索结果受用户偏好和用户经验影响。作者预见“推荐系统和检索系统的汇聚”，并且相信在搜索引擎算法上这些资源的整合可以产生用户的高满意度，让用户能在正确的时间收到正确的信息。

跟 IR 研究相关同时又阐述这方面研究的是：第 19 章针对上下文标签的检索问题；第 3 章介绍了根植于当前搜索引擎技术的基于上下文方法的概述。

最后，推荐系统最终目标是给用户提供了有用的信息，因此，在推荐最终能否被接受这个问题上，人机交互起了重大作用。实际上，许多领域的研究清楚地表明，从用户角度来看，HCI 中与系统可用性相关的方面对用户是否愿意使用推荐系统以及为了更有效的推荐结果而提供主动的输入有巨大的影响。这些问题在 1.6 节讨论过。

## 1.8 出现的问题和挑战

### 1.8.1 本书对出现的问题的讨论

从前面的讨论可以很明显地看出，推荐系统的研究正在向众多不同的方向发展，同时新的主题不断出现，或者正成为更重要的研究课题。读者也可以参考最近的 ACM RecSys 会议资料，参考其他优秀的论文，将其作为额外的研究素材[7, 3]。本手册中涵盖许多这种话题。实际上，很多已经介绍过了，例如，上下文感知推荐(第 7 章)；新的可视化技术(第 17 章)；基于社区的个性化搜索(第 18 章)；基于信任的推荐系统(第 20 章)。其他一些重要的话题在手册最后两部分，我们现在简短地介绍一下这些章节。

第 19 章介绍社会标签系统(STS)，这是随着 Web 2.0 的发展而出现的跟推荐系统相关的新话题。STS 如同 Flickr、Bibsonomy、Delicious 这类网站一样，允许普通用户发表和编辑内容，同样也允许用户产生和共享标签(即自定义关键字)。STS 用户正面临着信息过载问题，因为 STS 有数以百万计的用户，他们进入系统后发表的内容和标签不受控制，这就给传统的推荐系统造成检索困难。因此，需要推荐系统来帮助用户找到相关的信息，于是许多商业 STS 开始提供推荐服务(如 Delicious)。

该章讨论了 STS 推荐系统面对的新挑战，如推荐系统的新任务。这些不仅包括传统的内容推荐也包括相关标签推荐，甚至其他相似用户推荐。标签推荐系统(即推荐给用户一个物品的相近标签)，与传统的推荐系统有很多独特性，因为系统可以推荐经常出现的标签，不像传统的推荐系统一般不对相同物品进行二次推荐。另外，STS 的推荐系统解决的是三维问题(用户、资源、标签)，而不是传统推荐系统的二维问题(用户、物品)，这个问题会影响算法的复杂性。该章有一项最新调查，是关于支持 STS 的新一代推荐系统的。详述了给真实 STS 系统部署推荐系统遇到的挑战，并且为解决 STS 和标签推荐系统面临的挑战提供了新的算法。

第 21 章处理这样的情形，就是系统最好能够给一群用户推荐信息或物品，而不是给单个用户推荐。例如，推荐系统可以为一个群组选择电视节目，或者推荐歌单，其模型是基于整体群组成员的。为群组推荐很明显要比为个体推荐更复杂。假设我们能精确地知道

单个用户喜好,那么问题就是如何组合单个用户模型。在本章中,作者讨论了群体推荐如何工作,面临的问题,目前的进展。

第22章讨论了聚合用户偏好、准则和相似度这一普遍性问题。通常这类聚集是通过利用算术平均值或最大最小值函数完成的。但是许多其他有灵活性和适应性,并且最终能产生更相关的推荐结果的聚合函数经常被忽略。在本章,作者回顾了聚集函数的基础和特性,并且介绍了最重要的几个,包括广义集群,Choquet积分和Sugeno积分,有序加权平均,三角模自生成,双极性聚合函数。这种函数可以模拟输入条件之间的各种相互作用,包括连续的、分散的和混合的行为。

在第23章中,作者着重讲了推荐系统的另一个重要问题,即推荐系统生命周期过程中需要积极寻找新数据的问题。这个问题通常被忽略,因为用户访问系统时,系统就假设没有足够大的空间管理系统收集的数据(如评分)。实际上,推荐系统可以激发用户兴趣,许多系统在推荐过程中明确地询问用户偏好。因此,通过调整推荐过程,用户可以提供一系列不同的信息。特别是系统可以要求用户评价特定商品,用户对这些物品的评价在特定的条件下是非常有用的,比如,为一些用户或者整个用户群体产生更加多样化的推荐结果,或者简单地提高系统的预测准确性。此时,主动学习出现了;主动学习可以增强推荐系统,帮助用户更了解自己的偏好,产生更有效和更有用的问题。同时,主动学习可以为系统提供新信息,为随后的推荐作分析。因此,推荐系统应用主动学习能使推荐过程个性化[61]。这可以通过允许系统积极地影响用户遇到的物品(例如,在用户注册或用户定期使用时给用户展示物品)来完成,同样也可以通过用户自由探索自己的偏好来完成。

第24章介绍了另一个新兴话题,即多准则推荐系统。多数推荐系统通常是考虑单一标准值来决定物品的效用,例如,整体评价或者一个用户对一个物品的评分。但是目前这个假设被认为具有局限性。因为用户做选择的时候,系统为一个特定用户推荐的物品是否合适要根据用户考虑的多方面因素判断。多准则的组合可以影响用户意见,从而产生更有效更精确的推荐。

第24章提供了多准则推荐系统的一个概述。首先,它把推荐问题定义为一个多准则决策问题,并且回顾了支持多准则推荐实现的方法和技术。随后,集中讨论了多准则评分推荐技术的分类,这种技术主要是根据物品在不同准则上的评分向量对用户效用模型建模来产生推荐的。给出了有关利用多准则评分计算评分预测值并生成推荐的当前算法的概述。该章包括这些推荐系统的有争议问题和未来面临的挑战。

本书最后一章(第25章)概述了安全方面的话题。这个话题在过去几年是一个重要问题。目前这个话题的工作在[28, 45, 102, 112]有介绍。该章分析能产生更具健壮性推荐的算法设计,即推荐不易受到恶意用户影响。实际上,协同过滤推荐系统依赖用户信誉,即默认假设用户与系统相互作用的目的是为自己得到好的推荐,从而给邻居提供有用的数据。然而,在某些情况下用户与推荐系统的交互有一系列的目的,这些目的可能是与系统所有者或系统多数使用者背道而驰的。也就是说,这些用户想去破坏推荐系统网站或者干预提供给访客的推荐,例如,给一些物品更高的分数或更低的分数,而不是进行公正的评价。

该章作者提出一个有效的攻击模型,这种攻击的成本很低,但是对系统输出产生的影响很大。由于这些攻击可以在攻击网站时运行良好,因此很有必要检测它们并尽快制定对策。同时,研究人员已经研究出许多能经受攻击的健壮性算法,而且与有效攻击相比,这些算法具有较低的影响曲线。这些方法在本章中也涉及了。利用这些技术的结合,研究人



员考虑的不再是避免攻击而是去控制攻击的影响,让他们付出高昂的代价。

### 1.8.2 挑战

新兴的具有挑战性的推荐系统话题,不限于我们上面提到的章节中描述的那些。此外,覆盖所有的主题并不在这个简短介绍的范围内。读者可以参考本书最后的讨论部分,讨论其他值得关注的问题。

下面简短地介绍其他具有挑战性的话题,我们认为这些话题对推荐系统的研究发展很重要,但本书并没有涵盖这些内容。

- **大的真实数据集背景下的算法扩展性。**随着核心技术进步与成熟的研究,算法扩展性明显成为推荐系统的重要问题,它决定着如何把核心推荐技术嵌入真实系统里,如何处理用户和物品互动(评分、偏好、评论等)产生的大量动态的数据集。有个方法是在相对较小的数据集离线测试,但这个方法在非常大的数据集上可能失效或者完全不适应。需要进行新方法和大规模的评估的研究[91, 92, 33, 38, 116, 75, 75]。
- **主动推荐系统,**即虽然没有明确的请求[90, 24, 62, 80],推荐系统也产生推荐。到目前为止大多数的推荐系统都遵循一个“pull”模型[94],用户为获得推荐而发起请求。时过境迁,计算机无处不在,用户间总是相连的,人们很自然地想到推荐系统能够探测到隐藏的请求。所以需要预测不仅是推荐什么,还要考虑什么时候,如何去推送推荐。这样推荐系统变得积极主动而不用被视为骚扰。
- **推荐系统中的用户隐私保护**[81, 26, 79, 56, 17, 28, 102, 16, 5, 53, 70, 114]。推荐系统利用用户数据产生个性化推荐。在尝试建立更多更好的推荐系统时,它们尽可能多地收集用户数据。这无疑会对用户隐私产生负面影响,用户可能开始觉得系统知道太多关于自己真实想法的信息。因此,需要设计解决方案,简约并明智地使用用户数据。同时这些解决方案将确保用户信息库不被恶意用户随意获得。
- **推荐给目标用户的物品多样性**[104, 66, 69, 55, 54, 46, 119]。推荐列表中包含的物品如果具有一定程度的多样性,用户就更有可能找到合适的物品。为受限种类的产品做完美推荐是没有意义的,除非用户已经表示他的偏好在一个狭窄的范围内。有许多情况,特别是在推荐过程的早期阶段,用户想要探索新的多样性的方向。在这种情况下,用户使用推荐系统作为一个知识发现工具。在这个话题上的研究仍处于初级阶段,需要描述“多样性”的特性,即我们要从几次不同推荐中还是从一次推荐中寻找多样性,以及如何保证推荐精确性的同时兼有多样性。
- **在建立推荐列表的过程中整合用户的长期和短期偏好**[6, 40, 74]。推荐系统可以分为两类:一种是建立长期的配置资料,通过聚集所有系统收集的用户交易数据产生推荐(如系统过滤);另一种更侧重于用户的短期偏好,例如,在基于案例的方法中。显然,在解决偏好整合问题上,这两方面都很重要,无论是精确的用户任务还是物品的可用性都要兼顾。实际上,在当有足够的证据表明用户的短期偏好与长期偏好发生偏离时,需要建立混合模型来正确地决定是要发生兴趣偏移,还是不要偏向用户的偶然偏好。
- **通用的用户模型和交叉领域推荐系统**能够在不同的系统和应用领域调配用户数据[41, 18, 52, 19, 20, 49, 15]。使用通用的用户模型技术,一个单一的推荐系

统可以产生多种物品的推荐。这对于一般的推荐系统是不可能的,因为一般的推荐系统能够把多种技术整合到一个混合的方法中,但是却不能利用在一个领域收集到的用户偏好信息来在另一个领域产生推荐结果。

- **在开放式网络中运行的分布式推荐系统**[38, 116, 92, 113, 17, 102]。大多数推荐系统的计算模型是典型的客户端-服务器结构。用户客户端请求推荐,推荐系统服务端产生推荐以回应请求。这显然是个严重的限制,受到集成式控制系统中所有传统问题的折磨。新兴的网格计算或云计算的出现,为推荐系统实现更具健壮性和灵活性的计算模型提供了良机。
- **最优化推荐序列的推荐系统**[120, 99, 10, 59, 61, 107, 106]。我们已经提到过传统的推荐系统是为了提高推荐质量,这依赖一个简单的方法:一次性的请求/响应。会话式推荐系统可以通过实现学习能力进一步提高,不仅可以优化推荐的物品,而且可以优化如何使用户系统在所有可能情况下都能展开对话。
- **移动上下文的推荐系统**[117, 98, 55, 51, 4, 115, 111, 57, 29, 9, 77, 76, 89, 73, 44, 95, 13]。移动计算作为个人计算最自然的平台。当用户移动的时候可能产生推荐请求,例如,在城市旅游时的超市或者酒店。因此我们需要使用户界面“移动”起来(需求是“移动”的),需要设计能充分利用有限的移动资源(计算能力和屏幕尺寸)的计算方案。

最后,在结束引言之前,我们想介绍一些额外的挑战,即在最近的 RecSys 会议中一个教程中讨论的,该次会议地点在纽约,时间是 2009 年 10 月 22 日到 25 日, [<http://recsys.acm.org/tutorial3.pdf>]。John Riedl(Uni-versity of Minnesota)、Todd Beaupre(Yahoo!)和 John Sanders(Netflix)提出研究推荐系统的八个重要挑战:透明度、探索与利用、指导准则、时间价值、用户行为解释、评价推荐系统、扩展性、学术/工业之间关系。

在引言中已经讨论过一些问题。例如,当讨论解释推荐的作用时介绍了透明度,我们强调透明度在使推荐更易被用户接受上有重大作用。同时,推荐系统的评估方法(也就是评估中被测量的一系列重要角度)这个话题在另一章(第 8 章)有完整的阐述。

推荐的时间价值在上下文感知推荐系统(第 7 章)的评论部分讨论过。然而,实际上面临的挑战是,给定的一系列推荐不可能永远适用,而这些物品需要过一段时间才能被系统推荐。例如,当谈及新闻物品时,人们显然想要了解最新发生的事件和新闻,而这些新闻甚至是与最初发布间隔一天后仍不能被推荐给用户。

主动学习(第 23 章)对探索还是利用的话题有所涉及。这个过程面临的挑战也是设计师必须妥善解决的主要困境,即在系统当前可用数据的给定条件下,推荐目前能识别出的好物品,还是为了得到未来更新也可能更好的推荐结果,而去探索用户的偏好(比如,要求用户对额外的特定物品评分)。

索引式导航栏,导航栏指的是将传统的推荐列表(即推荐结果)和能够让用户在选项中更加自主浏览的工具相结合。用户行为解读指的是,除了用户显式评分之外,还可以探测、分析用户在操作推荐系统中产生的更多行为,并用于构建更好的预测模型。这个想法是,每个用户的用户行为都应该被利用在推荐过程中。但是解读用户行为(即一个行为的背后意图)是个挑战,有些行为需要被舍弃,因为这不是由原来真正的用户产生的,例如,在同一个浏览器中不同用户执行的行为,或者是虚假和恶意注册,或者是由机器人和爬虫引起的数据或日志。

先前也提到过可扩展性。我们强调,很显然这是个问题,当前文献中对这个问题的探

讨并不完整, 因为可扩展性一直被大多数工业实践者研究调查。

最后, 关于工业界和学术界合作一事在研讨会上讨论得兴致勃勃。工业界有一些特殊的问题, 但这些问题没有明确地公开。这有很多原因, 包括不向竞争对手透露关键信息。相反, 学术界正在寻找能在允许的时间和资源框架内解决的问题, 学术界通常会专注一个能对科学界产生影响的专题。这种不同会使工业/学术合作变得困难。但是推荐系统是个需要新的实际挑战的领域, 这些挑战是有助于解决问题或使问题变得成熟的。如果我们不解决那些有用的但是有风险的挑战, 那么该领域的研究就真正有停滞不前的风险。

我们希望本书能成为实践工作者和研究人员的有用工具, 有助于在这个令人激动并且有用的研究领域进一步开发知识。这样我们相信能减少两组人员(工业与学术)走不同路线的风险。目前推荐系统领域极大地得益于工业界和学术界对此的兴趣和努力。因此, 当他们两组人阅读本书时, 我们都会奉上最美好的祝愿。希望能吸引更多的研究者在这个非常有趣又充满挑战的领域奋斗!

## 参考文献

1. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.* **23**(1), 103–145 (2005)
2. Adomavicius, G., Tuzhilin, A.: Personalization technologies: a process-oriented perspective. *Commun. ACM* **48**(10), 83–90 (2005)
3. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* **17**(6), 734–749 (2005)
4. Ahn, H., Kim, K.J., Han, I.: Mobile advertisement recommender system using collaborative filtering: Mar-cf. In: *Proceedings of the 2006 Conference of the Korea Society of Management Information Systems*, pp. 709–715 (2006)
5. Aïmeur, E., Brassard, G., Fernandez, J.M., Onana, F.S.M.: Alambic : a privacy-preserving recommender system for electronic commerce. *Int. J. Inf. Sec.* **7**(5), 307–334 (2008)
6. Aïmeur, E., Vézeau, M.: Short-term profiling for a case-based reasoning recommendation system. In: R.L. de Mántaras, E. Plaza (eds.) *Machine Learning: 2000, 11th European Conference on Machine Learning*, pp. 23–30. Springer (2000)
7. Anand, S.S., Mobasher, B.: Intelligent techniques for web personalization. In: *Intelligent Techniques for Web Personalization*, pp. 1–36. Springer (2005)
8. Arazy, O., Kumar, N., Shapira, B.: Improving social recommender systems. *IT Professional* **11**(4), 38–44 (2009)
9. Averjanova, O., Ricci, F., Nguyen, Q.N.: Map-based interaction with a conversational mobile recommender system. In: *The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBIComm '08*, pp. 212–218 (2008)
10. Baccigalupo, C., Plaza, E.: Case-based sequential ordering of songs for playlist recommendation. In: T. Roth-Berghofer, M.H. Göker, H.A. Güvenir (eds.) *ECCBR, Lecture Notes in Computer Science*, vol. 4106, pp. 286–300. Springer (2006)
11. Bailey, R.A.: *Design of comparative experiments*. Cambridge University Press Cambridge (2008)
12. Balabanovic, M., Shoham, Y.: Content-based, collaborative recommendation. *Communication of ACM* **40**(3), 66–72 (1997)
13. Bellotti, V., Begole, J.B., hsin Chi, E.H., Ducheneaut, N., Fang, J., Isaacs, E., King, T.H., Newman, M.W., Partridge, K., Price, B., Rasmussen, P., Roberts, M., Schiano, D.J., Walendowski, A.: Activity-based serendipitous recommendations with the magitti mobile leisure guide. In: M. Czerwinski, A.M. Lund, D.S. Tan (eds.) *CHI*, pp. 1157–1166. ACM (2008)
14. Ben-Shimon, D., Tsikinovsky, A., Rokach, L., Meisels, A., Shani, G., Naamani, L.: Recommender system from personal social networks. In: K. Wegrzyn-Wolska, P.S. Szczepaniak (eds.) *AWIC, Advances in Soft Computing*, vol. 43, pp. 47–55. Springer (2007)
15. Berkovsky, S.: *Mediation of User Models: for Enhanced Personalization in Recommender Systems*. VDM Verlag (2009)

16. Berkovsky, S., Borisov, N., Eytani, Y., Kuflik, T., Ricci, F.: Examining users' attitude towards privacy preserving collaborative filtering. In: International Workshop on Data Mining for User Modeling, at User Modeling 2007, 11th International Conference, UM 2007, Corfu, Greece, June 25, 2007, Proceedings (2007)
17. Berkovsky, S., Eytani, Y., Kuflik, T., Ricci, F.: Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In: RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems, pp. 9–16. ACM Press, New York, NY, USA (2007)
18. Berkovsky, S., Kuflik, T., Ricci, F.: Cross-technique mediation of user models. In: Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems [AH2006], pp. 21–30. Dublin (2006)
19. Berkovsky, S., Kuflik, T., Ricci, F.: Mediation of user models for enhanced personalization in recommender systems. *User Modeling and User-Adapted Interaction* **18**(3), 245–286 (2008)
20. Berkovsky, S., Kuflik, T., Ricci, F.: Cross-representation mediation of user models. *User Modeling and User-Adapted Interaction* **19**(1-2), 35–63 (2009)
21. Billsus, D., Pazzani, M.: Learning probabilistic user models. In: UM97 Workshop on Machine Learning for User Modeling (1997). URL <http://www.dfki.de/~bauer/um-ws/>
22. Bridge, D., Göker, M., McGinty, L., Smyth, B.: Case-based recommender systems. *The Knowledge Engineering review* **20**(3), 315–320 (2006)
23. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction* **6**(2-3), 87–129 (1996)
24. Bulander, R., Decker, M., Schiefer, G., Kolmel, B.: Comparison of different approaches for mobile advertising. *Mobile Commerce and Services*, 2005. WMCS '05. The Second IEEE International Workshop on pp. 174–182 (2005)
25. Burke, R.: Hybrid web recommender systems. In: *The Adaptive Web*, pp. 377–408. Springer Berlin / Heidelberg (2007)
26. Canny, J.F.: Collaborative filtering with privacy. In: *IEEE Symposium on Security and Privacy*, pp. 45–57 (2002)
27. Carenini, G., Smith, J., Poole, D.: Towards more conversational and collaborative recommender systems. In: *Proceedings of the 2003 International Conference on Intelligent User Interfaces*, January 12–15, 2003, Miami, FL, USA, pp. 12–18 (2003)
28. Cheng, Z., Hurley, N.: Effective diverse and obfuscated attacks on model-based recommender systems. In: RecSys '09: Proceedings of the third ACM conference on Recommender systems, pp. 141–148. ACM, New York, NY, USA (2009)
29. Church, K., Smyth, B., Cotter, P., Bradley, K.: Mobile information access: A study of emerging search behavior on the mobile internet. *ACM Trans. Web* **1**(1), 4 (2007)
30. Cosley, D., Lam, S.K., Albert, I., Konstant, J.A., Riedl, J.: Is seeing believing? how recommender system interfaces affect users' opinions. In: *Proceedings of the CHI 2003 Conference on Human factors in Computing Systems*. Fort Lauderdale, FL (2003)
31. Felfernig, A., Friedrich, G., Schubert, M., Mandl, M., Mairitsch, M., Teppan, E.: Plausible repairs for inconsistent requirements. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI'09)*, pp. 791–796. Pasadena, California, USA (2009)
32. Fisher, G.: User modeling in human-computer interaction. *User Modeling and User-Adapted Interaction* **11**, 65–86 (2001)
33. George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: *Proceedings of the 5th IEEE Conference on Data Mining (ICDM)*, pp. 625–628. IEEE Computer Society, Los Alamitos, CA, USA (2005)
34. Golbeck, J.: Generating predictive movie recommendations from trust in social networks. In: *Trust Management, 4th International Conference, iTrust 2006, Pisa, Italy, May 16–19, 2006, Proceedings*, pp. 93–104 (2006)
35. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. *Commun. ACM* **35**(12), 61–70 (1992)
36. Groh, G., Ehlig, C.: Recommendations in taste related domains: collaborative filtering vs. social filtering. In: GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work, pp. 127–136. ACM, New York, NY, USA (2007)
37. Guy, I., Zwerdling, N., Carmel, D., Ronen, I., Uziel, E., Yorgev, S., Ofek-Koifman, S.: Personalized recommendation of social software items based on social relations. In: RecSys '09: Proceedings of the third ACM conference on Recommender systems, pp. 53–60. ACM, New York, NY, USA (2009)
38. Han, P., Xie, B., Yang, F., Sheng, R.: A scalable p2p recommender system based on distributed collaborative filtering. *Expert systems with applications* (2004)

39. Hayes, C., Cunningham, P.: Smartradio-community based music radio. *Knowledge Based Systems* **14**(3-4), 197-201 (2001)
40. He, L., Zhang, J., Zhuo, L., Shen, L.: Construction of user preference profile in a personalized image retrieval. In: *Neural Networks and Signal Processing, 2008 International Conference on*, pp. 434-439 (2008)
41. Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., von Wilamowitz-Moellendorff, M.: Gumo - the general user model ontology. In: *User Modeling 2005, 10th International Conference, UM 2005, Edinburgh, Scotland, UK, July 24-29, 2005, Proceedings*, pp. 428-432 (2005)
42. Herlocker, J., Konstan, J., Riedl, J.: Explaining collaborative filtering recommendations. In: *In proceedings of ACM 2000 Conference on Computer Supported Cooperative Work*, pp. 241-250 (2000)
43. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Transaction on Information Systems* **22**(1), 5-53 (2004)
44. Horozov, T., Narasimhan, N., Vasudevan, V.: Using location for personalized POI recommendations in mobile environments. In: *Proc. Int'l Sym. Applications on Internet*, pp. 124-129. IEEE Computer Society (2006)
45. Hurley, N., Cheng, Z., Zhang, M.: Statistical attack detection. In: *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pp. 149-156. ACM, New York, NY, USA (2009)
46. Hwang, C.S., Kuo, N., Yu, P.: Representative-based diversity retrieval. In: *Innovative Computing Information and Control, 2008. ICICIC '08. 3rd International Conference on*, pp. 155-155 (2008)
47. Jannach, D.: Finding preferred query relaxations in content-based recommenders. In: *3rd International IEEE Conference on Intelligent Systems*, pp. 355-360 (2006)
48. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: *Recommender Systems An Introduction*. Cambridge University Press (2010)
49. Jessenitschnig, M., Zanker, M.: A generic user modeling component for hybrid recommendation strategies. *E-Commerce Technology, IEEE International Conference on* **0**, 337-344 (2009). DOI <http://doi.ieeecomputersociety.org/10.1109/CEC.2009.83>
50. Kay, J.: Scrutable adaptation: Because we can and must. In: *Adaptive Hypermedia and Adaptive Web-Based Systems, 4th International Conference, AH 2006, Dublin, Ireland, June 21-23, 2006, Proceedings*, pp. 11-19 (2006)
51. Kim, C.Y., Lee, J.K., Cho, Y.H., Kim, D.H.: Viscors: A visual-content recommender for the mobile web. *IEEE Intelligent Systems* **19**(6), 32-39 (2004)
52. Kobsa, A.: Generic user modeling systems. In: P. Brusilovsky, A. Kobsa, W. Nejdl (eds.) *The Adaptive Web, Lecture Notes in Computer Science*, vol. 4321, pp. 136-154. Springer (2007)
53. Kobsa, A.: Privacy-enhanced personalization. In: D. Wilson, H.C. Lane (eds.) *FLAIRS Conference*, p. 10. AAAI Press (2008)
54. Koren, Y., Bell, R.M., Volinsky, C.: Matrix factorization techniques for recommender systems. *IEEE Computer* **42**(8), 30-37 (2009)
55. Kramer, R., Modsching, M., ten Hagen, K.: Field study on methods for elicitation of preferences using a mobile digital assistant for a dynamic tour guide. In: *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pp. 997-1001. ACM Press, New York, NY, USA (2006)
56. Lam, S.K., Frankowski, D., Riedl, J.: Do you trust your recommendations? an exploration of security and privacy issues in recommender systems. In: G. Müller (ed.) *ETRICS, Lecture Notes in Computer Science*, vol. 3995, pp. 14-29. Springer (2006)
57. Lee, H., Park, S.J.: Moners: A news recommender for the mobile web. *Expert Systems with Applications* **32**(1), 143 - 150 (2007)
58. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* **7**(1), 76-80 (2003)
59. Mahmood, T., Ricci, F.: Towards learning user-adaptive state models in a conversational recommender system. In: A. Hinneburg (ed.) *LWA 2007: Lernen - Wissen - Adaption*, Halle, September 2007, Workshop Proceedings, pp. 373-378. Martin-Luther-University Halle-Wittenberg (2007)
60. Mahmood, T., Ricci, F.: Improving recommender systems with adaptive conversational strategies. In: C. Cattuto, G. Ruffo, F. Menczer (eds.) *Hypertext*, pp. 73-82. ACM (2009)
61. Mahmood, T., Ricci, F., Venturini, A., Höpken, W.: Adaptive recommender systems for travel planning. In: W.H. Peter O'Connor, U. Gretzel (eds.) *Information and Communication Technologies in Tourism 2008, proceedings of ENTER 2008 International Conference*, pp. 1-11. Springer, Innsbruck (2008)



62. Mahmoud, Q.: Provisioning context-aware advertisements to wireless mobile users. *Multimedia and Expo, 2006 IEEE International Conference on* pp. 669–672 (2006)
63. Manning, C.: *Introduction to Information Retrieval*. Cambridge University Press, Cambridge (2008)
64. Massa, P., Avesani, P.: Trust-aware collaborative filtering for recommender systems. In: *Proceedings of the International Conference on Cooperative Information Systems, CoopIS*, pp. 492–508 (2004)
65. McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P.: Group recommender systems: a critiquing based approach. In: C. Paris, C.L. Sidner (eds.) *IUI*, pp. 267–269. ACM (2006)
66. McGinty, L., Smyth, B.: On the role of diversity in conversational recommender systems. In: A. Aamodt, D. Bridge, K. Ashley (eds.) *ICCBR 2003, the 5th International Conference on Case-Based Reasoning*, pp. 276–290. Trondheim, Norway (2003)
67. McGinty, L., Smyth, B.: Adaptive selection: An analysis of critiquing and preference-based feedback in conversational recommender systems. *International Journal of Electronic Commerce* 11(2), 35–57 (2006)
68. McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pp. 1097–1101. ACM Press, New York, NY, USA (2006)
69. McSherry, D.: Diversity-conscious retrieval. In: S. Craw, A. Preece (eds.) *Advances in Case-Based Reasoning, Proceedings of the 6th European Conference on Case Based Reasoning, ECCBR 2002*, pp. 219–233. Springer Verlag, Aberdeen, Scotland (2002)
70. McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the net. In: *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 627–636. ACM, New York, NY, USA (2009)
71. Mirzadeh, N., Ricci, F.: Cooperative query rewriting for decision making support and recommender systems. *Applied Artificial Intelligence* 21, 1–38 (2007)
72. Montaner, M., López, B., de la Rosa, J.L.: A taxonomy of recommender agents on the internet. *Artificial Intelligence Review* 19(4), 285–330 (2003)
73. Nguyen, Q.N., Ricci, F.: Replaying live-user interactions in the off-line evaluation of critique-based mobile recommendations. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 81–88. ACM Press, New York, NY, USA (2007)
74. Nguyen, Q.N., Ricci, F.: Conversational case-based recommendations exploiting a structured case model. In: *Advances in Case-Based Reasoning, 9th European Conference, ECCBR 2008, Trier, Germany, September 1–4, 2008. Proceedings*, pp. 400–414 (2008)
75. Papagelis, M., Rousidis, I., Plexousakis, D., Theoharopoulos, E.: Incremental collaborative filtering for highly-scalable recommendation algorithms. In: M.S. Hacid, N.V. Murray, Z.W. Ras, S. Tsumoto (eds.) *ISMIS, Lecture Notes in Computer Science*, vol. 3488, pp. 553–561. Springer (2005)
76. Park, M.H., Hong, J.H., Cho, S.B.: Location-based recommendation system using bayesian user's preference model in mobile devices. In: J. Indulska, J. Ma, L.T. Yang, T. Ungerer, J. Cao (eds.) *UIC, Lecture Notes in Computer Science*, vol. 4611, pp. 1130–1139. Springer (2007)
77. Park, S., Kang, S., Kim, Y.K.: A channel recommendation system in mobile environment. *Consumer Electronics, IEEE Transactions on* 52(1), 33–39 (2006). DOI 10.1109/TCE.2006.1605022
78. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review* 13, 393–408 (1999)
79. Polat, H., Du, W.: Privacy-preserving collaborative filtering using randomized perturbation techniques. In: *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003)*, 19–22 December 2003, Melbourne, Florida, USA, pp. 625–628 (2003)
80. Puerta Melguizo, M.C., Boves, L., Deshpande, A., Ramos, O.M.: A proactive recommendation system for writing: helping without disrupting. In: *ECCE '07: Proceedings of the 14th European conference on Cognitive ergonomics*, pp. 89–95. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1362550.1362569>
81. Ramakrishnan, N., Keller, B.J., Mirza, B.J., Grama, A., Karypis, G.: When being weak is brave: Privacy in recommender systems. *IEEE Internet Computing* cs.CG/0105028 (2001)
82. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing. In: *Advances in Case-Based Reasoning, 7th European Conference, ECCBR 2004, Madrid, Spain, August 30 - September 2, 2004, Proceedings*, pp. 763–777 (2004)
83. Reilly, J., Zhang, J., McGinty, L., Pu, P., Smyth, B.: Evaluating compound critiquing recom-



- menders: a real-user study. In: EC '07: Proceedings of the 8th ACM conference on Electronic commerce, pp. 114–123. ACM, New York, NY, USA (2007)
84. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: An open architecture for collaborative filtering of netnews. In: Proceedings ACM Conference on Computer-Supported Cooperative Work, pp. 175–186 (1994)
  85. Resnick, P., Varian, H.R.: Recommender systems. *Communications of the ACM* **40**(3), 56–58 (1997)
  86. Ricci, F.: Travel recommender systems. *IEEE Intelligent Systems* **17**(6), 55–57 (2002)
  87. Ricci, F., Cavada, D., Mirzadeh, N., Venturini, A.: Case-based travel recommendations. In: D.R. Fesenmaier, K. Woeber, H. Werthner (eds.) *Destination Recommendation Systems: Behavioural Foundations and Applications*, pp. 67–93. CABI (2006)
  88. Ricci, F., Missier, F.D.: Supporting travel decision making through personalized recommendation. In: C.M. Karat, J.O. Blom, J. Karat (eds.) *Designing Personalized User Experiences in eCommerce*, pp. 231–251. Kluwer Academic Publisher (2004)
  89. Ricci, F., Nguyen, Q.N.: Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems* **22**(3), 22–29 (2007). DOI <http://doi.ieeecomputersociety.org/10.1109/MIS.2007.43>
  90. Sae-Ueng, S., Pinyapong, S., Ogino, A., Kato, T.: Personalized shopping assistance service at ubiquitous shop space. *Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008. 22nd International Conference on* pp. 838–843 (2008). DOI 10.1109/WAINA.2008.287
  91. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Incremental singular value decomposition algorithms for highly scalable recommender systems. In: *Proceedings of the 5th International Conference in Computers and Information Technology* (2002)
  92. Sarwar, B.M., Konstan, J.A., Riedl, J.: Distributed recommender systems for internet commerce. In: M. Khosrow-Pour (ed.) *Encyclopedia of Information Science and Technology* (II), pp. 907–911. Idea Group (2005)
  93. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: *The Adaptive Web*, pp. 291–324. Springer Berlin / Heidelberg (2007)
  94. Schafer, J.B., Konstan, J.A., Riedl, J.: E-commerce recommendation applications. *Data Mining and Knowledge Discovery* **5**(1/2), 115–153 (2001)
  95. Schifanella, R., Panisson, A., Genà, C., Ruffo, G.: Mobhinter: epidemic collaborative filtering and self-organization in mobile ad-hoc networks. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 27–34. ACM, New York, NY, USA (2008)
  96. Schwartz, B.: *The Paradox of Choice*. ECCO, New York (2004)
  97. van Setten, M., McNee, S.M., Konstan, J.A.: Beyond personalization: the next stage of recommender systems research. In: R.S. Amant, J. Riedl, A. Jameson (eds.) *IUI*, p. 8. ACM (2005)
  98. van Setten, M., Pokraev, S., Koolwaaij, J.: Context-aware recommendations in the mobile tourist application compass. In: W. Nejdl, P. De Bra (eds.) *Adaptive Hypermedia 2004*, pp. 235–244. Springer Verlag (2004)
  99. Shani, G., Heckerman, D., Brafman, R.I.: An mdp-based recommender system. *Journal of Machine Learning Research* **6**, 1265–1295 (2005)
  100. Sharda, N.: *Tourism Informatics: Visual Travel Recommender Systems, Social Communities, and User Interface Design*. Information Science Reference (2009)
  101. Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "word of mouth". In: *Proceedings of the Conference on Human Factors in Computing Systems (CHI'95)*, pp. 210–217 (1995)
  102. Shokri, R., Pedarsani, P., Theodorakopoulos, G., Hubaux, J.P.: Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In: *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pp. 157–164. ACM, New York, NY, USA (2009)
  103. Sinha, R.R., Swearingen, K.: Comparing recommendations made by online systems and friends. In: *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries* (2001)
  104. Smyth, B., McClave, P.: Similarity vs diversity. In: *Proceedings of the 4th International Conference on Case-Based Reasoning*. Springer-Verlag (2001)
  105. Swearingen, K., Sinha, R.: Beyond algorithms: An HCI perspective on recommender systems. In: J.L. Herlocker (ed.) *Recommender Systems, papers from the 2001 ACM SIGIR Workshop*. New Orleans, LA - USA (2001)

106. Taghipour, N., Kardan, A.: A hybrid web recommender system based on q-learning. In: Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), Fortaleza, Ceara, Brazil, March 16-20, 2008, pp. 1164–1168 (2008)
107. Taghipour, N., Kardan, A., Ghidary, S.S.: Usage-based web recommendations: a reinforcement learning approach. In: Proceedings of the 2007 ACM Conference on Recommender Systems, RecSys 2007, Minneapolis, MN, USA, October 19-20, 2007, pp. 113–120 (2007)
108. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Scalable collaborative filtering approaches for large recommender systems. *J. Mach. Learn. Res.* **10**, 623–656 (2009)
109. Tan, P.N.: Introduction to Data Mining. Pearson Addison Wesley, San Francisco (2006)
110. Thompson, C.A., Goker, M.H., Langley, P.: A personalized system for conversational recommendations. *Artificial Intelligence Research* **21**, 393–428 (2004)
111. Tung, H.W., Soo, V.W.: A personalized restaurant recommender agent for mobile e-service. In: S.T. Yuan, J. Liu (eds.) Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service, EEE'04, pp. 259–262. IEEE Computer Society Press, Taipei, Taiwan (2004)
112. Van Roy, B., Yan, X.: Manipulation-resistant collaborative filtering systems. In: RecSys '09: Proceedings of the third ACM conference on Recommender systems, pp. 165–172. ACM, New York, NY, USA (2009)
113. Wang, J., Pouwelse, J.A., Lagendijk, R.L., Reinders, M.J.T.: Distributed collaborative filtering for peer-to-peer file sharing systems. In: H. Haddad (ed.) SAC, pp. 1026–1030. ACM (2006)
114. Wang, Y., Kobsa, A.: Performance evaluation of a privacy-enhancing framework for personalized websites. In: G.J. Houben, G.I. McCalla, F. Pianesi, M. Zancanaro (eds.) UMAP, *Lecture Notes in Computer Science*, vol. 5535, pp. 78–89. Springer (2009)
115. Wietsma, R.T.A., Ricci, F.: Product reviews in mobile decision aid systems. In: Pervasive Mobile Interaction Devices (PERMID 2005) - Mobile Devices as Pervasive User Interfaces and Interaction Devices - Workshop in conjunction with: The 3rd International Conference on Pervasive Computing (PERVASIVE 2005), May 11 2005, Munich, Germany, pp. 15–18. LMU Munich (2005)
116. Xie, B., Han, P., Yang, F., Shen, R.: An efficient neighbor searching scheme of distributed collaborative filtering on p2p overlay network. *Database and Expert Systems Applications* pp. 141–150 (2004)
117. Yuan, S.T., Tsao, Y.W.: A recommendation mechanism for contextualized mobile advertising. *Expert Systems with Applications* **24**(4), 399–414 (2003)
118. Zhang, F.: Research on recommendation list diversity of recommender systems. *Management of e-Commerce and e-Government, International Conference on* pp. 72–76 (2008)
119. Zhang, M.: Enhancing diversity in top-n recommendation. In: RecSys '09: Proceedings of the third ACM conference on Recommender systems, pp. 397–400. ACM, New York, NY, USA (2009)
120. Zhou, B., Hui, S., Chang, K.: An intelligent recommender system using sequential web access patterns. In: Cybernetics and Intelligent Systems, 2004 IEEE Conference on, vol. 1, pp. 393–398 vol.1 (2004)
121. Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: WWW '05: Proceedings of the 14th international conference on World Wide Web, pp. 22–32. ACM Press, New York, NY, USA (2005)



# 基础技术

- 第 2 章 推荐系统中的数据挖掘方法
- 第 3 章 基于内容的推荐系统：前沿和趋势
- 第 4 章 基于近邻推荐方法综述
- 第 5 章 协同过滤算法的高级课题
- 第 6 章 开发基于约束的推荐系统
- 第 7 章 情境感知推荐系统

# 推荐系统中的数据挖掘方法

Xavier Amatriain、Alejandro Jaimes、Nuria Oliver 和 Josep M. Pujol

**摘要** 本章概述了推荐系统中用到的一些重要的数据挖掘技术。首先描述的是常见的数据预处理方法，如抽样和降维。其次，回顾推荐系统中最重要分类技术，包括贝叶斯网络和支持向量机。我们对 K-means 聚类算法进行描述并讨论其几个替代算法。我们也会介绍有效训练过程中的关联规则和相关算法。除了介绍这些技术，我们也会考察它们在推荐系统中的使用情况，同时介绍它们成功应用的案例。

## 2.1 简介

推荐系统典型地运用了其他相邻领域的技术和方法，如人机交互和信息检索。但是，大多数系统的核心算法都可以理解成数据挖掘技术的一个特例。

数据挖掘的过程一般由三个连续执行的步骤组成：**数据预处理**[59]、**数据分析和结果解释**(见图 2.1)。我们将在 2.2 节中分析一些最重要的数据预处理方法。鉴于数据抽样、数据降维、距离函数在推荐系统中的意义及所担任的重要角色，我们将特别关注这些内容。从 2.3 节到 2.5 节，将总体介绍在推荐系统中最常使用的数据挖掘方法：分类、聚类、关联规则发现(图 2.1 详细显示了本章中包含的不同主题)。

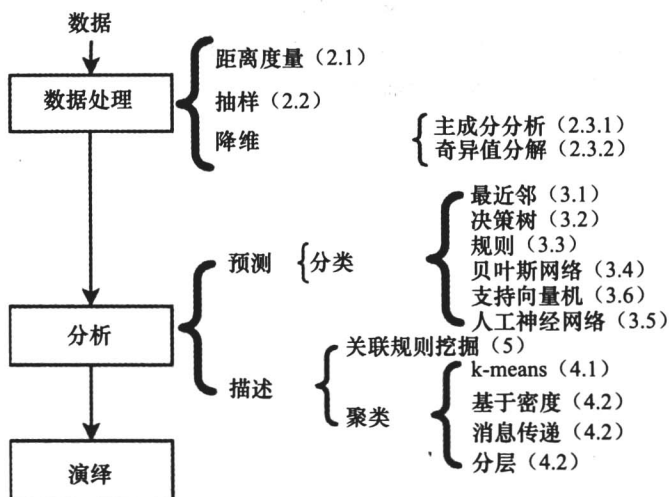


图 2.1 数据挖掘中主要的步骤和方法及其对应的章节

Xavier Amatriain, Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain e-mail: xar@tid.es  
 Alejandro Jaimes, Yahoo! Research, Av. Diagonal, 177, Barcelona 08018, Spain. Work on the chapter was performed while the author was at Telefonica Research. e-mail: ajames@yahoo-inc.com  
 Nuria Oliver, Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain e-mail: nuriao@tid.es  
 Josep M. Pujol, Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain e-mail: jmps@tid.es  
 翻译: 吉林大学-丁彬钊 审核: 刘湘勇, 胡聪(胡户主), 郑州大学-吴宾



本章不会完整回顾数据挖掘方法,而是强调数据挖掘算法在推荐系统领域中的影响,并概述已成功应用的主要数据挖掘技术。感兴趣的读者可以进一步参考数据挖掘课本(见[28, 73]),或参考贯穿全章的引文。

## 2.2 数据预处理

我们把数据定义为一组对象及其属性的集合,其中属性定义为性质或者是对象的特征。对象的其他名称包括记录、物品、得分、样本、观察值或者实例。属性也可以称为变量、字段、特性或者特征。

真实数据通常需要经过预处理,以便于机器学习技术在分析阶段使用。本节紧紧围绕推荐系统设计三个尤为重要的问题展开。首先,我们回顾不同的相似度,或者距离度量方式。其次,我们需要讨论抽样问题,一种可以减少大数据集中物品数量并保持其主要特征的方法。最后,我们将阐述降维方法中最常用的技术。

### 2.2.1 相似度度量方法

协同过滤推荐备受青睐方法之一是使用 kNN 分类,我们将在 2.3.1 节中讨论。这种分类技术(如同大多数的分类和聚类技术)主要取决于定义合适的相似度或者距离度量方法。

最简单、最常用的距离度量是欧几里得距离:

$$d(x, y) = \sqrt{\sum_{k=1}^n (x_k - y_k)^2} \quad (2.1)$$

其中,  $n$  是维数(属性数),  $x_k$  和  $y_k$  分别是数据对象  $x$  和  $y$  的第  $k$  个属性值(分量)。

闵可夫斯基距离是欧几里得距离的推广:

$$d(x, y) = \left( \sum_{k=1}^n |x_k - y_k|^r \right)^{\frac{1}{r}} \quad (2.2)$$

其中,  $r$  是距离的度(参数)。取决于  $r$  值的不同,一般的闵可夫斯基距离有专用的名称:

$r=1$ , 城市街区(也叫曼哈顿、出租车、L1 范数)距离。

$r=2$ , 欧几里得距离(L2 范数)。

$r=\infty$ , 上确界( $L_{\max}$  或  $L_{\infty}$  范数), 这是任意维度对象属性间的最大距离。

马氏距离定义如下:

$$d(x, y) = \sqrt{(x - y)\sigma^{-1}(x - y)^T} \quad (2.3)$$

其中,  $\sigma$  是数据的协方差矩阵。

另一个常用的方法是把物品看作  $n$  维空间的文档向量,并且计算它们相似度作为形成夹角的余弦值,其公式如下:

$$\cos(x, y) = \frac{(x \cdot y)}{\|x\| \|y\|} \quad (2.4)$$

其中,  $\cdot$  表示向量的点积,  $\|x\|$  是向量  $x$  的长度。这个相似度称为余弦相似度或者是 L2 范数。

物品之间的相似度还可以用他们的相关度计算,用以度量对象间的线性关系。尽管有几个相关系数可能被应用,但皮尔逊相关性系数是最常用的。给出点  $x$  和  $y$  的协方差及它们的标准差  $\sigma$ , 我们用以下公式计算皮尔逊相关性:

$$\text{Pearson}(x, y) = \frac{\sum(x, y)}{\sigma_x \times \sigma_y} \quad (2.5)$$

推荐系统一般会使用余弦相似度式(2.4)或者是皮尔逊相关性(或者它们的许多变种方法中的一种,如加权方案)。第4和5章详述在协同过滤中不同距离函数的使用。但是,前面提到的大部分其他距离度量方法都可能用到。Spertus等[69]在Orkut社交网络的环境中做了大规模的研究来评估六种不同的相似度度量方法。尽管由于实验的特殊设置,结果会有偏差,但有趣的是余弦相似度是其中效果最好的度量方法。Lathia等[48]也做了一些相似度度量的研究,其总结,在一般的案例中,推荐系统的预测精确性不受相似度度量方法选择的影响。事实上,在他们的工作中,使用随机的相似度度量有时会产生比使用已知任何众所周知的方法更好的结果。

最后,在一些只有二进制属性的物品案例中,可以采用几个相似度度量方法。首先,计算M01、M10、M11和M00数量,其中M01代表 $x$ 是0同时 $y$ 是1这个属性的数量,M10代表 $x$ 是1同时 $y$ 是0这个属性的数量,以此类推。根据这些数值我们可以计算得到:简单匹配系数  $\text{SMC} = \frac{\text{number of matches}}{\text{number of attributes}} = \frac{\text{M11} + \text{M00}}{\text{M01} + \text{M10} + \text{M00} + \text{M11}}$ ; Jaccard 系数  $\text{JC} = \frac{\text{M11}}{\text{M01} + \text{M10} + \text{M11}}$ 。广义 Jaccard(Tanimoto)系数,是JC关于连续值属性或计数属性的一个变型,计算为  $d = \frac{x \cdot y}{\|x\|^2 + \|y\|^2 - x \cdot y}$ 。

### 2.2.2 抽样

抽样是数据挖掘从大数据集中选择相关数据子集的主要技术。它用于数据预处理和最终解释步骤中。之所以使用抽样是因为处理全部数据集的计算开销太大。它也可以被用来创建训练和测试数据集。这个情况下,训练集被用于分析阶段学习参数或配置算法,而测试集被用来评估训练阶段获得的模型或者配置,确保它在将来产生的未知数据上运行良好。

抽样的关键是发现具有整个原始数据集代表性的子集,也就是说,其具有与整个数据集大概类似的兴趣属性。最简单的抽样技术是随机抽样,任意物品被选中的概率相同。但也有更复杂的方法。例如,在分层抽样中数据基于特殊特征被分成几个部分,之后对每个部分独立进行随机抽样。

最常用的抽样方法包含使用无取代的抽样:当物品被选择的时候,物品被从整体中取走。但是,执行取代抽样也是可以的,物品即使被选择也不用从整体中去除,允许同样的样本被选择多次。

在分离训练集和测试集时,通常做法是使用80/20的训练集和测试集比例,并使用无替代的标准随机抽样。这意味着我们使用无替代随机抽样方法去选择20%的实例为测试集,把剩下的80%进行训练。80/20的比例应该作为一个经验规则,一般来说,超过2/3的任何值作为训练集是合适的。

抽样可能导致过特殊化划分的训练和测试数据集。因此,训练过程可以重复好几次。从原始数据集中创建训练集和测试集,使用训练数据进行模型训练并且使用测试集中的样例进行测试。接下来,选择不同的训练/测试集进行训练/测试过程,这个过程会重复 $K$ 次。最后,给出 $K$ 次学习模型的平均性能。这个过程是著名的交叉验证。交叉验证技术有很多种。在重复随机样本中,标准的随机抽样过程要执行 $K$ 次。在 $n$ 折交叉校验中,

数据集被分成  $n$  份。其中一份被用来测试模型, 剩下  $n-1$  份被用来进行训练。交叉验证过程重复  $n$  次,  $n$  个子样本中每一个子样本都只使用一次作为验证数据。最后, 留一法(LOO)可以看作  $n$  折交叉验证的极端例子, 其中  $n$  被设置为数据集中物品的数量。因此, 算法运行许多次而每次数据点只使用其中一个作为测试。我们需要注意的是, 正如 Isaksson 等讨论的那样[44], 除非数据集足够大, 否则交叉验证可能不可信。

在推荐系统中常用的方法是从用户中抽取可用的反馈以用户评分的形式来划分训练和测试。交叉验证的方法同样也很常见。尽管在一般的案例中标准随机抽样是可接受的, 但是在其他场景中我们需要用不同的方法定向调整抽样出来的测试集。例如, 我们可能决定只抽样最近的评分数据, 因为这些是现实情况下我们需要预测的。我们可能还有兴趣确保每个用户的评分比例被保存在测试集, 因此需要对每一个用户使用随机抽样。然而, 所有这些涉及评估推荐的问题仍是一个探讨和研究点。

### 2.2.3 降维

推荐系统中不仅有定义高维空间特征的数据集, 而且在空间中信息非常稀疏, 例如, 每个对象就那么几个有限的特征有值。密度, 以及点之间的距离, 这些对于聚类和孤立点检测非常重要, 但在高维空间中的意义并不大。这就是著名的维度灾难。降维技术通过把原始高维空间转化成低维有助于克服这类问题。

稀疏和维度灾难是推荐系统中反复出现的问题。即使在最简单的背景下, 我们很可能都会有成千上万行的行和列稀疏矩阵, 其中大部分值是零。因此, 降低维度就自然而然了。应用降维技术带来这样的结果, 其也可以直接适用于计算推荐的预测值, 即它可以作为推荐系统设计的方法, 而不仅是数据预处理技术。

接下来, 我们概述两个在推荐系统中最相关的降维算法: 主成分分析(PCA)和奇异值分解(SVD)。这些技术可以单独作为推荐方法使用, 或作为在本章提到的其他任何技术的预处理步骤。

#### 2.2.3.1 主成分分析

主成分分析(PCA)[45]是一种经典统计方法, 用来发现高维数据集的模式。主成分分析可以获得一组有序的成分列表, 其根据最小平方误差计算出变化最大的值。列表中第一个成分所代表的变化量要比第二个成分所代表的变化量大, 以此类推。我们可以通过忽略这些对变化贡献较小的成分来降低维度。

图 2.2 显示了通过高斯合并产生的二维点云中的 PCA 分析结果。数据集中之后, 主要成分由  $u_1$  和  $u_2$  来表示。考虑到新坐标轴的长度所涉及的能量被包含在它们的特征向量中。因此, 对于图 2.2 中列举的特殊例子, 第一个成分  $u_1$  占能量的 83.5%, 这意味着移除第二个成分  $u_2$  将只失去 16.5% 的信息。根据经验规则选择  $m'$  以便于累计能量超过一定的阈值, 一般是 90%。PCA 允许我们把数据投影到新的坐标系中来重新表示原始数据矩阵:  $X'_{n \times m'} = X_{n \times m} W'_{m \times m'}$ 。新的数据矩阵  $X'$  降低了  $m-m'$  维度并保证包含大部分的原始数据  $X$  的信息。

PCA 是一种强大的技术, 但也有重要的限制。PCA 依赖于以线性合并为基础的经验数据集, 尽管一般的非线性 PCA 方法已经提出。PCA 的另一个重要假设是原始数据集是从高斯分布中抽取出来的。当这个假设不正确时, 就无法保证主要成分的有效性。

尽管目前的趋势似乎表明其他的矩阵分解技术更受欢迎, 如 SVD 或者非负矩阵分解,

但是早期用得最多还是 PCA。Goldberg 等在在线笑话推荐的内容中提出使用 PCA 方法 [37]。他们的系统，著名的 Eigentaste，开始于标准的用户评分矩阵。然后从所有用户都评分过的 item 里选出一个子集作为测试集。这个新矩阵被用来计算全局相关矩阵，这些矩阵使用了标准的二维 PCA。

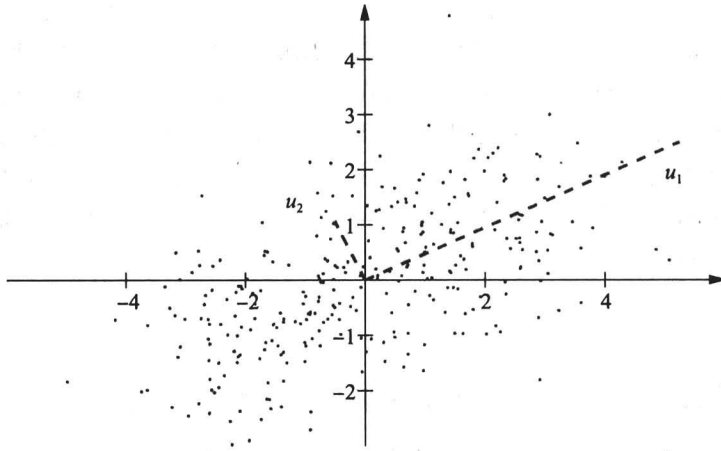


图 2.2 PCA 基于高斯合并的二维点云。使用 PCA 得到的主成分是  $u_1$  和  $u_2$ ，其长度与到包含在所有成分的能量相关

### 2.2.3.2 奇异值分解

奇异值分解 [38] 是一个强大的降维工具。它是矩阵分解方法的特殊实现，因此它也和 PCA 相关。在 SVD 分解中的关键问题是发现低维特征空间，这些新特征代表概念以及在集合内容中的每一个概念强度都是可计算的。因为 SVD 可以自动获取到低维空间上的语义概念，它可以被用来当作潜在语义分析的基础，潜在语义分析 [24] 是一种在信息检索中非常受欢迎的文本分类技术。

SVD 的核心算法基于以下的理论：把矩阵  $A$  分解成  $A = U\lambda V^T$  是可行的。给出  $n \times m$  矩阵的数据  $A$  ( $n$  个物品， $m$  个特征)，我们可以获得一个  $n \times r$  的矩阵  $U$  ( $n$  个物品， $r$  个概念)，一个  $r \times r$  的对角矩阵  $R$  (概念的长度)，以及  $m \times r$  的矩阵  $V$  ( $m$  特征， $r$  概念)。图 2.3 阐述了这个想法。 $R$  的对角矩阵包含奇异值，其总是为正并且是降序排列。 $U$  矩阵可以解释成物品概念相似矩阵，矩阵  $V$  是特征概念相似性矩阵。

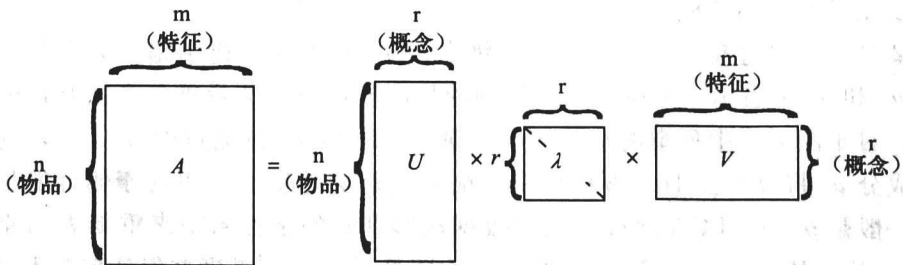


图 2.3 阐述了最基本的 SVD：物品  $\times$  特征矩阵可以分解成三个不同的部分：物品  $\times$  概念、概念强度、概念  $\times$  特征

为了计算矩形矩阵  $A$  的 SVD，我们考虑如下公式  $AA^T$  和  $A^TA$ 。 $U$  的列是  $AA^T$  的特征向量， $V$  的列是  $A^TA$  的特征向量。矩阵对角线上的奇异值是  $AA^T$  和  $A^TA$  非零特征值的平方根。因此，为了计算矩阵  $A$  的 SVD，首先计算  $AA^T(T)$  以及  $A^TA(D)$ ，然后计算  $T$

和  $D$  的特征向量和特征值。

在  $\lambda$  中的特征值  $r$  是有序递减的。因此, 初始矩阵  $A$  可以通过截取前  $k$  个特征值来近似构造。截取的 SVD 构造了一个近似矩阵  $A$  的  $k$  秩矩阵  $A_k = U_k \lambda_k V^T$ 。  $A_k$  是最近似原始矩阵的  $K$  秩矩阵。最近似表达的是最小化  $A$  与  $A_k$  元素之间的平方差之和。被截取的 SVD 代表降维成  $k$  维空间后的潜在结构, 这一般意味着特征向量中的噪声被降低。

使用 SVD 作为工具来提高协同过滤已经有一段时间了。Sarwar 等[66]在论文中描述了使用 SVD 的两种不同方法。首先, SVD 可以用来发现用户与产品之间的潜在关系。为了完成这个目的, 他们首先用物品平均评分值去填充用户-物品矩阵的 0 值项, 然后通过减去用户对所有物品平均评分来正规化这些矩阵。这些矩阵用 SVD 来分解, 其分解结果在一些细微的操作之后可以直接用来计算预测值。其他方法是使用从 SVD 中提取出的低维空间中的结果来提高在 kNN 方法的邻居信息。

正如 Sarwar 等[65]描述的那样, SVD 的一大优势是有增量算法来计算近似的分解。这使得我们在接收到新用户或者是评分的时候, 没有必要重新计算用先前存在的数据构建的模型。同样的想法后来被 Brand[14]的在线 SVD 模型扩充和正式采纳。在成功应用到 Netflix Prize 之后, 增量 SVD 方法的使用最近已经成为常用的方法。Simon Funk 的简单增量 SVD 方法的发表被标志为竞赛中的转折点[35]。自从它发表之后, 在该领域已经发表了几篇改进的 SVD(详细信息可以参考 Paterek 的全部 SVD 的算法[56], 或者是 Kurucz 等的 SVD 参数评估[47])。

最后, 应该注意的是矩阵分解(MF)的不同变化方法, 如非负的矩阵分解(NNMF)已经被使用[74]。本质上来说, 这些算法类似于 SVD。最基本的想法是把评分矩阵分解成两个部分: 一个部分包含描述用户的特征, 另一个部分包含描述物品的特征。矩阵分解通过引入偏项到模型中来处理缺失数据比 SVD 方法要好。但是, SVD 方法中也可以在预处理阶段通过用物品的平均值来取代零值来处理。需要注意的是 SVD 和 MF 都可能产生过拟合的问题。但是已存在改进的 MF, 如正规化内核矩阵分解, 能有效地避免这个问题。MF 和 SVD 方法的主要问题是, 由于计算的复杂性每次数据升级更新时重新计算分解是不现实的。但是, Rendle 和 Schmidt-Thieme[62]提出一种在线的方法允许不用重新计算所有整个模型来更新分解近似值。

第 5 章会详细介绍在 Netflix Prize 的环境中 SVD 和 MF 的使用, 是对本章简介的详细补充。

## 2.2.4 去噪

数据挖掘中采集的数据可能会有各种噪声, 如缺失数据, 或者是异常数据。去噪是非常重要的预处理步骤, 其目的是在最大化信息量时去除掉不必要的影响。

在一般意义上我们把噪声定义为在数据收集阶段收集到的一些可能影响数据分析和解释结果的伪造数据。在推荐系统的环境中, 我们区分自然的和恶意的噪声[55]。前者提到的噪声是用户在选择偏好反馈时无意产生的。后者是为了偏离结果在系统中故意引入的。

很显然恶意的噪声能够影响推荐的输出。但是, 我们的研究推断正常的噪声对推荐系统性能的影响是不可忽略的[4]。为了解决这个问题, 我们设计了一个去噪方法, 能够通过要求用户重新评价一些物品来提高精确度[5]。我们推断通过预处理步骤来提高精确度能够比复杂的算法优化效果要好得多。



## 2.3 分类

分类器是从特征空间到标签空间的映射，其中特征代表需要分类的元素的属性，标签代表类别。例如，餐厅推荐系统能够通过分类器来实现，其分类器基于许多特征描述把餐厅分成两类中的一类(好的，不好的)。

有许多种类型的分类器，但是一般情况下我们谈的有监督分类器和无监督分类器。在有监督分类器中，我们预先知道一组标签或是类别，并且我们有一组带有标签的数据，用来组成训练集。在无监督分类中，类别都是提前未知的，其任务是恰当地组织好我们手中的元素(按照一些规则)。在本节中我们描述几个算法来学习有监督分类，无监督分类(如聚类)将在 2.4 节中进行描述。

### 2.3.1 最近邻

基于样本的分类(instance-based classifier)通过存储训练记录并使用它们来预测未知样本的标签类别。一个常见的例子是所谓的死记硬背学习(roter-learner)。这种分类器记住了所有的训练集，并且只有在新记录的属性与训练集中样本完全匹配时才会分类。一个更加精确和通用的基于样本的分类是近邻分类(kNN)。给出一个要分类的点，kNN 分类器能够从训练记录中发现  $k$  个最近的点。然后按照它最近邻的类标签来确定所属类标签。算法的基本思想是，如果一个样本落入由一个类标签主导的领域，是因为这个样本可能属于这个类。

假设我们需要确定样本  $q$  的类别  $l$ ，定义训练集是  $x = \{\{x_1, l_1\} \cdots \{x_n, l_n\}\}$ ，其中  $x_j$  是第  $j$  个元素， $l_j$  是它的类标签， $k$  的最近邻可以找到子集  $Y = \{\{y_1, l_1\} \cdots \{y_k, l_k\}\}$ ，使得  $Y \in X$  且  $\sum_{k=1}^k d(q, y_k)$  是最下限。 $Y$  包含  $X$  中的  $k$  个离  $q$  最近的样本点。那么， $q$  的类标签是  $l = f(\{l_1 \cdots l_k\})$ 。

也许在 kNN 中最具有挑战的问题是如何选择  $k$  的值。如果  $k$  太小，分类可能对噪声点太敏感。但是如果  $k$  太大，近邻范围可能会包含其他类中太多的点。图 2.4 右图展示了不同的  $k$  值下最终确定不同的类标签。 $k=1$  时类标签可能是圆形的，而  $k=7$  时类标签是正方形。注意到例子中的查询点正好处于两个类别中的边界上，因此，分类很困难。

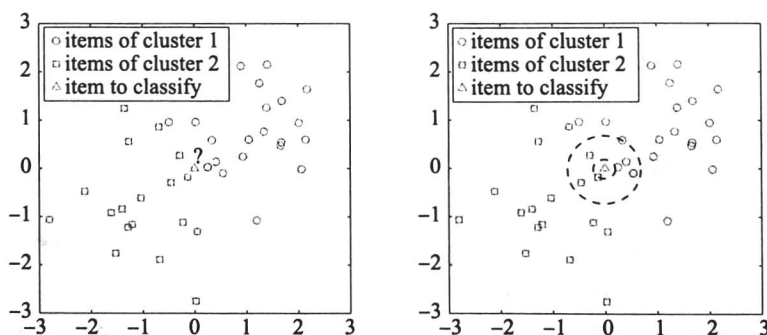


图 2.4  $k$  近邻的例子。左边的子图显示带有两个类标签的训练点(圆形和正方形)和查询点(三角形)。右边的子图阐述  $k=1$  和  $k=7$  时的最近邻。查询点按照简单多数规则，当  $k=1$  时被分类为正方形，当  $k=5$  时被分类为圆形。注意查询点正好在两个类别之间的边界线上

kNN 分类器在所有的机器学习的算法中是最简单的。因为 kNN 不要建立一个显示的模型，所以被认为是一个懒的学习者。不像饥饿学习者，如决策树或是基于规则的系统

(分别参见 2.3.2 节和 2.3.3 节), kNN 分类器把许多的决策留给了分类的步骤。因此, 分类未知记录的花费相当大。

近邻算法是 CF 最常用的一种方法, 因而被用来设计推荐系统。事实上, 任何的推荐系统综述, 如 Adomavicius 和 Tuzhilin 的那篇[1], 都会包含本书所提到的近邻使用的简介。这种分类的优点之一是它的概念和 CF 很相关: 发现志趣相投的用户(或者是类似的物品)实质上等价于发现给定用户或者是物品的邻居。其他的优势是: 作为 kNN 分类器这样一个懒惰学习者, 它不需要学习和维持一个给定的模型。因此, 在原则上, 系统能够适应用户评分矩阵的急速变化。遗憾的是, 这是以重新计算邻居和相似矩阵为代价的。这也是我们要提出一种用精简后的专家集合来挑选邻居模型的原因[3]。

尽管 kNN 方法简单和直观, 但是它的结果精确, 非常易于提升。事实上, 它对于协同推荐的实际标准的主导地位最近才被基于降维的方法所挑战, 如 2.2.3 节所叙述的。也就说, 针对协同过滤方法的传统的 kNN 方法已经在几个方向上得到了提升。例如, 在 Netflix Prize 的实验环境中, Bell 和 Koren 建议一种方法来移除全局的影响, 如一些物品可能会吸引用户一致给低分。他们提出邻居建立时立即计算插入权值的优化算法。

详见第 5 章和第 4 章基于邻居使用改进 CF 技术的更多细节。

### 2.3.2 决策树

决策树[61, 63]是以树结构形式对目标属性(或类)进行分类的分类器。要分类的观察数据(或物品)是由属性及其目标值组成的。树的节点可以是: a) 决策节点, 在这些节点中一个简单属性值被测试来决定应用哪一个子树; b) 叶子节点指示目标属性的值。

对于决策树归纳有许多的算法: 最常提到的是搜索算法, 包括 CART、ID3、C4.5、SLIQ、SPRINT。递归搜索算法(最早的也是最容易理解的算法)依赖作用于给定属性的测试条件, 通过它们的目标值来区别这些观察值。算法一旦找到测试条件推导出的划分区域, 就会反复迭代, 直到划分区域为空, 或者观察数据都有相同的目标值。

拆分可以通过最大的信息增益来决定, 定义如下:

$$\Delta_i = I(\text{parent}) - \sum_{j=1}^{k_i} \frac{N(v_j)I(v_j)}{N} \quad (2.6)$$

其中,  $k_i$  是属性  $i$  的值,  $N$  是观察数据的数量,  $v_j$  是根据属性  $i$  的值得到的观察值的第  $j$  个划分。最后,  $I$  是衡量不纯节点的函数。有各种不同的不纯衡量方法: 基尼指数、熵、误分类误差是在文献中最常用的。

一旦所有的观察值属于同一个类(或者是在连续属性中的相同范围), 决策树的推导就结束。这表明叶子节点的非纯度是零。然而, 因为实际的原因, 大部分的决策树通过剪枝技术实现, 如果节点的非纯度或者观察值的数量低于某个阈值, 节点不再进行分裂。

使用决策树建立一个分类器的主要优点是, 构建代价比较小并且在分类未知的对象方面速度比较快。与其他基础的分类技术相比, 决策树另一个好的方面是在维持精度的同时, 它产生的一系列规则容易被解释(见 2.3.3 节)。

推荐系统中的决策树可以用在基于模型的方法里。一种可能是用内容特征建立决策树模型, 对描述用户偏好的所有变量建模。Bouza 等[12]利用这种想法, 使用物品可用的语义信息构建一个决策树。用户只评价两个物品之后就能构造决策树。每个物品的特征被用来建立一个解释用户评分的模型。他们使用每一个特征的信息增益作为分裂准则。需要注意的是, 尽管这种方法从理论视角看很有趣, 但是在他们系统上报告的精确性比推荐平均

评分的方法要差。

正如可以预料到的那样,建立一个试图解释决策过程中所有参数的决策树是非常困难以及不现实的。但是,决策树可以被用来模拟系统的一个特殊部分。例如,Cho等[8]提出一个结合关联规则(见2.5节)和决策树的在线购物推荐系统。决策树被用来作为一个过滤器来选择哪些用户可以作为推荐的目标。为了建立这个模型,他们创建了一个候选用户集,用户集是在给定的时间帧内从一个给定的目录下选了商品的这些用户。在他们的案例中,选择作为构造决策树的因变量是用户是否会在相同的分类下再买新的产品。Nikovski和Kulev[54]随后提出一个与之类似的结合决策树和关联规则的方法。在他们的方法中,先是在购买的数据集中发现频繁物品集,然后应用标准的树学习算法来简化推荐规则。

在推荐系统中另一个使用决策树的选择是使用它们作为物品排序的工具。使用决策树来排序已经在一些环境下被研究,而且很明确都是为了这个目的[7, 17]。

### 2.3.3 基于规则的分类

基于规则分类器是通过一组“if...then...”的规则集合划分数据。规则的前提或条件是属性连词的表达式。规则的结论是一个正或者负的分类。

如果对象的属性满足规则的条件,可以说规则 $R$ 覆盖对象 $x$ 。我们定义规则的覆盖性为满足前提的部分记录。另一方面,我们定义准确性为既满足前提又满足结论的部分记录。如果规则彼此之间是独立的,我们说分类器包含互斥的规则,例如,每一个记录最多被一个规则覆盖。最后,如果属性值的所有可能组合都被覆盖,例如,一个记录至少被一个规则覆盖,则认为分类器具有详尽规则(exhaustive rules)。

为了建立一个基于规则的分类器,我们可以用从数据中直接抽取规则的直接方法。这种方法的例子是RIPPER或CN2。另一方面,使用间接的方法从其他分类模型中抽取规则很常见,例如,决策树模型或神经模型。

基于规则分类器的优点是它们表示很明确,因为它们是符号化的并且可以在没有任何转化的情况下操作数据的属性。基于规则的分类器,由决策树扩充,容易解释,容易生成,并且它们能有效地分类新的对象。

但是,与决策树方法类似,建立一个完整基于规则的推荐模型是很难的。事实上,这种方法在推荐的环境中不是很流行,因为得到一个基于规则的系统意味着我们要么具有一些决策过程中的显式的先验知识,要么从另一个模型中提取规则,如决策树。但是基于规则的系统通过注入一些领域知识或商业规则来提高推荐系统的性能。例如,Anderson等[6]实现了一个协同音乐推荐系统,这个系统通过应用一个基于规则的系统在协同过滤的结果中提高性能。如果用户给某个音乐家的专辑评分很高,那么这个音乐家的其他专辑的预测评分也会提高。

Gutta等[29]实现了一个关于电视内容的基于规则的推荐系统。为此,他们首先使用C4.5决策树,然后分解成规则来分类电视节目。Basu等[9]利用归纳的方法使用Ripper[20]系统从数据中学习规则。在他们的报告中,使用混合内容和协同数据来学习规则的结果明显好于单纯的CF方法。

### 2.3.4 贝叶斯分类器

贝叶斯分类器[34]是解决分类问题的一个概率框架。它基于条件概率定义和贝叶斯理

论。贝叶斯统计学派使用概率来代表从数据中学习到的关系的不确定性。此外,先验的概念非常重要,因为它们代表了我们的期望值,或者真正关系可能是什么的先验知识。特别的是,给定数据后,模型的概率(后验概率)是和似然值乘以先验概率的乘积成比例的。似然值部分包含了数据的影响,而先验概率则表明观测数据之前模型的可信度。

贝叶斯分类器把每一个属性和类标签当作随机变量(连续或者离散)。给定一个带有  $N$  个属性的记录  $(A_1, A_2, \dots, A_N)$ , 目标是预测类  $C_k$ , 方法是在给定数据  $P(C_k | A_1, A_2, \dots, A_N)$  下, 找到能够最大化该类后验概率的  $C_k$  的值。应用贝叶斯理论  $P(C_k | A_1, A_2, \dots, A_N) \propto P(A_1, A_2, \dots, A_N | C_k)P(C_k)$ 。

一个特殊但是最常用的分类器是朴素贝叶斯分类器。为了估计条件概率,  $P(A_1, A_2, \dots, A_N | C_k)$ 。假设属性的概率独立, 比如, 一个特殊属性的存在与否和其他任何的属性的存在与否没有关系。这种假设导致  $P(A_1, A_2, \dots, A_N | C_k) = P(A_1 | C_k)P(A_2 | C_k) \dots P(A_N | C_k)$ 。

朴素贝叶斯的主要好处是, 受孤立噪声点和不相关的属性的影响小, 并且在概率估算期间可以通过忽略实例来处理缺失值。但是, 独立性假设对一些相互关联的属性来说可能不成立。在这种情况下, 通常的方法是使用所谓的贝叶斯信念网络(或简称贝叶斯网络)。BBN 使用非循环图表示属性之间的依赖性, 并使用概率表表示节点与直接父亲之间的联系。与朴素贝叶斯分类器方法类似, BBN 可以很好地处理不完整的数据, 对于模型的过拟合有相当的健壮性。

贝叶斯分类器在基于模型的推荐系统中特别受欢迎。它们经常被用来为基于内容的推荐生成模型。当然, 它们也被用于协同环境中。例如, Ghani 和 Fano[36], 使用朴素贝叶斯实现了一个基于内容的推荐系统。使用这个模型允许在百货商店环境中从不相关的目录中推荐产品。

Miyahara 和 Pazzani[52]实现了一个基于朴素贝叶斯分类器的推荐系统。为了达到这个目的定义了两个类: 喜欢和不喜欢。在这种的环境中他们提出两个方法来使用朴素贝叶斯分类器: 数据转化模型假设所有的特征都是完全独立的, 特征选择作为一个预处理步骤来实施。另一方面, 稀疏数据模型假设只有已知的特征是对分类有益的信息。此外, 当估算概率的时候, 只使用用户都共同评价的数据。实验显示两种模型性能好于基于相关性的 CF。

Pronk 等[58]用贝叶斯朴素聚类器作为基础来合并用户组件并且提高性能, 特别是冷启动环境。为了做到这一点, 他们提出给每个用户维持两个属性文件, 一个从历史评分中学习得到, 另一个由用户显式地创建。两种分类器的混合可以通过这样的方式来控制, 在早期阶段没有太多历史评分时采用用户定义属性文件, 然后在随后的阶段再用学习型分类器取而代之。

在 2.3.3 节中我们提到 Gutta 等[29]在电视内容上实现了一个基于规则方法的推荐系统。此外他们还实验了贝叶斯分类器。首先定义一个两类分类器, 类别包括: 看过和没看过。用户配置文件是属性的集合, 以及他们作为正样本和负样本出现的次数。这会被用来计算节目属于某个特定分类的先验概率, 以及当节目是正向或负向时, 某个给定特征会出现的条件概率。在这样案例中, 必须注意到的是特征涉及内容(类型)和环境(时间)。新节目的后验概率是从这些(环境和内容)中计算得来的。

Breese 等[15]实现了将每个节点关联到每个物品的贝叶斯网络。状态与每个可能的投票值相关。在网络中, 每一个物品将有一组父亲节点作为它最好的预测器。条件概率表被

决策树取代。作者报告显示在几组数据集上这种模型的结果比几种近邻算法的结果要好。

分层的贝叶斯网络也用在一些环境中, 用作信息过滤添加领域知识的方法[78]。但是分层的贝叶斯网络的问题之一是, 当其中的用户过多时, 学习和升级模型的代价非常大。Zhang 和 Koren[79]提出一个标准期望最大化模型的变种, 能够在基于内容推荐系统中的环境中加速这种过程。

### 2.3.5 人工神经网络

人工神经网络(ANN)[81]由一组内连接点和带权链接组成, 其想法来自于生物大脑的结构。ANN 中的节点称为神经元, 类似于生物神经。这些简单的功能单元组成网络, 网络在用有效数据训练之后能够学习分类问题。

ANN 的最简单模型是感知器模型, 如图 2.5 所示。如果我们把激活函数  $\phi$  特指为简单的阈值函数, 则输出就是根据每条链接的权重将输入值累加, 然后和某个阈值  $\theta_k$  相比较。输出函数可以由式(2.7)来表达。感知模型是具有简单和有效学习算法的线性聚类器。但是, 除了使用在感知模型中的阈值函数, 还有几种其他对于激活函数通用的选择, 如多层感知机、正切双曲或者是阶梯函数。

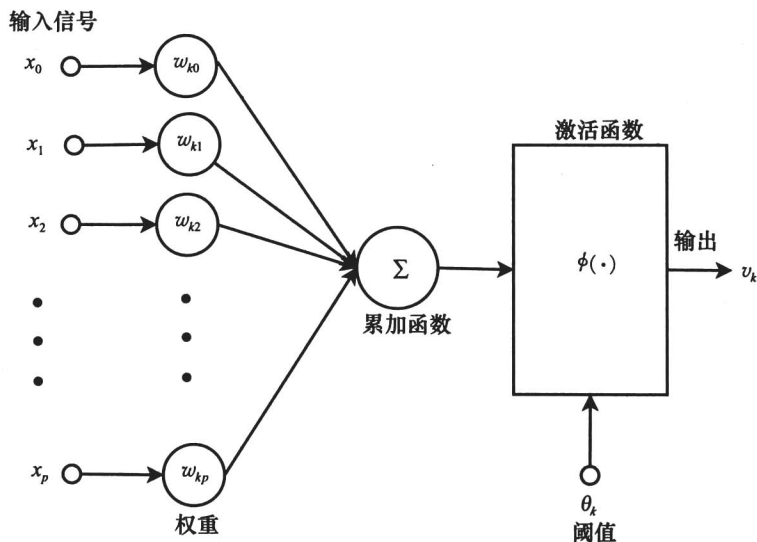


图 2.5 感知器模型

$$y_k = \begin{cases} 1, & \text{若 } \sum x_i w_{ki} \geq \theta_k \\ 0, & \text{若 } \sum x_i w_{ki} < \theta_k \end{cases} \quad (2.7)$$

ANN 可以有许多的层。在 ANN 中的层被分成三种类型: 输入、隐藏、输出。输入层的单元响应进入网络的数据。隐藏层接受从输入单元中的带权输出。输出层响应隐藏层中的带权输出并且产生最终的网络输出。使用神经元作为原子功能单元, 在网络中有许多种可能的架构来把它们结合在一起。但是, 最常用的方法是使用前馈 ANN。在这个例子中, 信号严格在一个方向传播: 从输入到输出。

ANN 最主要的优点是(取决于激活函数)能做非线性的分类任务, 并且由于并行属性, 它们高效甚至能够在部分网络受损的情况下操作。主要的缺点是, 它很难对给定的问题提供理想的网络拓扑, 并且一旦确定拓扑后它的表现水平就会位于分类错误率的下限。ANN 属于一种次符号分类器, 也就是说, 在推理知识的时候不提供任何语义知识, 说白



了这是一种黑盒方法。

ANN 能够以类似于贝叶斯网络的方法用来构建基于模型的推荐系统。但是, 没有令人信服的研究表明 ANN 是否会提升性能。事实上, Pazzani 和 Billsus[57]做了一个综合实验, 使用几种机器学习算法进行网页推荐。他们的主要目标是比较朴素贝叶斯分类器与计算开销更大的候选方法, 如决策树和神经网络。他们的实验结果显示决策树的效果明显不好。他们推断似乎没有必要用非线性分类器, 如 ANN。Berka 等[31]使用 ANN 为网页导航建立 URL 推荐系统。他们实现了专门基于访问路径而与内容无关的系统, 比如, 把域名和访问它们的人数关联起来。为此, 他们使用了后向传播算法训练的前馈多层感知器。

ANN 可以被用来结合(或是混合)几个推荐模块或者数据源中的输入。例如, Hsu 等[30]建立一个电视推荐模型, 通过四个不同的源导入数据: 用户的配置文件和自身看法、观看社区、节目元数据、观看环境。他们用后向传播算法来训练三层神经网络。Christakou 和 Stafylopatis[19]也建立了一个混合的基于内容的协同过滤推荐系统。基于内容的推荐系统在实现时对每个用户采用了三种神经网络, 其中每一个对应如下的一个特征: 种类、星级、摘要。他们使用弹性反向传播方法来训练 ANN。

### 2.3.6 支持向量机

支持向量机分类[23]的目标是发现数据的线性超平面(决策边界), 以边界最大化的方式分离数据。例如, 如果我们在二维平面上看两类分离的问题, 如图 2.6 阐述的那样, 很容易观察到分成两个类有许多种可能的边界线。每一个边界线都有一个相关的边缘。SVM 后面的理论支持是, 如果我们选择间隔最大化化的那一个, 将来对未知的物品分类出错的可能性就越小。

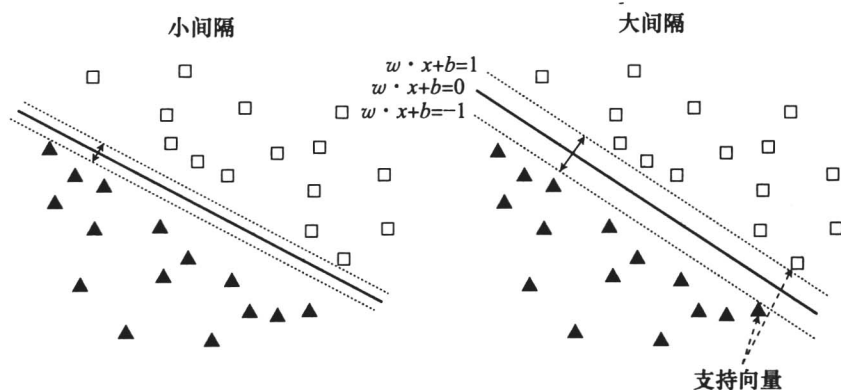


图 2.6 在二维上不同的边界决定可能分成不同的类。每一个边界有一个相关的间隔

两个类中的线性分离是通过函数  $w \cdot x + b = 0$  来实现的。我们定义能够划分物品类 +1 或 -1 的函数, 只要这些物品是被来自类划分函数的某个最小距离分开的。式(2.8)给出了相应的公式。

$$f(x) = \begin{cases} 1, & \text{若 } w \cdot x + b \geq 1 \\ -1, & \text{若 } w \cdot x + b \leq -1 \end{cases} \quad (2.8)$$

$$\text{边缘} = \frac{2}{\|w\|^2} \quad (2.9)$$

根据 SVM 的主要原理, 我们想要最大化两个类之间的间隔, 由式(2.9)给出。事实上

这等价于在给定  $f(x)$  的约束条件下, 最小化  $L(w) = \frac{\|w\|^2}{2}$  的倒数。这其实是带约束最优化的问题, 有许多数学方法可以解决它(如二次规划)。

如果物品不是线性分离的, 则可以通过引入一个松弛变量来把 SVM 转变为软间隔分类器。在这种情况下, 式(2.10)的最小化受限于式(2.11)新的  $f(x)$  定义。另一方面, 如果决策边界是非线性的, 我们需要转换数据到高维的空间。这个转换的完成得益于名为内核技巧的数学变换。最基本的想法是通过内核函数取代在式(2.8)中的点积。对于内核函数有许多不同的可行的选择, 比如多项式或多层感知器。但是最常用的内核函数是径向基函数系列(RBF)。

$$L(w) = \frac{\|w\|^2}{2} + C \sum_{i=1}^N \epsilon \quad (2.10)$$

$$f(x) = \begin{cases} 1, & \text{若 } w \cdot x + b \geq 1 - \epsilon \\ -1, & \text{若 } w \cdot x + b \leq -1 + \epsilon \end{cases} \quad (2.11)$$

支持向量机最近已经在许多的环境中获得较好的性能和效率。在推荐系统里, SVM 最近也显示出了显著效果。比如, Kang 和 Yoo[46]报告了一个实验研究, 其目的在于为基于 SVM 的推荐系统选择最好的预处理技术预测缺失值。他们特别使用了 SVD 和支持向量回归(SVR)。支持向量机推荐系统首先通过二进制化可用用户偏好数据的 80 个等级来建立。他们设置了几组实验, 并且报告了阈值为 32 时的最好结果, 例如, 32 和更小的值被分为喜欢, 较高值被分为不喜欢。用户的 ID 被用作分类的标签, 正负值被表达成偏好值 1 和 2。

Xu 和 Araki[76]用 SVM 建立一个电视节目推荐系统。他们用电子节目向导(Electronic Program Guide, EPG)的信息作为特征。但是为了减少特征, 他们移除了最低频次的单词。此外, 为了评价不同的方法, 他们用布尔值和词频率-逆文档频率(TFIDF)来衡量特征结构的权重。在前者, 0 和 1 被用来代表在内容中物品的缺失或出现。在后者则变成词频率-逆文档频率数值。

Xia 等[75]提出用不同的方法来使用 SVM 在 CF 环境中做推荐。他们探索平滑 SVM(SSVM)的使用。他们也介绍了一个基于 SSVM 的启发式方法, 来迭代估算在用户物品矩阵中的缺失元素。他们通过为每一个用户创建一个分类器来计算预测值。实验结果显示, 与 SSVM 以及传统的基于用户和基于物品的 CF 相比, SSVMBH 实验结果最好。最后, Oku 等[27]为情景感知推荐系统提出情景感知 SVM(C-SVM)的使用方法。他们比较了标准 SVM、C-SVM 和一种既使用 CF 又使用 C-SVM 的扩展算法。结果显示在餐厅推荐中情景感知方法最有效。

### 2.3.7 分类器的集成

使用分类器集成背后的最基本的思想是, 从训练数据构造一系列的分类器, 并通过聚集预测值来预测类标签。只要我们能假设这些分类器都是独立的, 分类器集成就有效。在这种情况下, 我们可以确定分类器产生的最糟糕的结果与在集成中的最坏分类是一样的。因此, 结合具有相似的分类错误的独立分类器将只会提升结果。

为了产生集成, 有几种可能的方法。最常用的两个技术是 Bagging 和 Boosting。在 Bagging 方法中, 我们采用带替换的抽样, 在每一个自举样本(bootstrap sample)上建立分类。每一个样本被选择的概率是  $\left(1 - \frac{1}{N}\right)^N$ , 如果  $N$  足够大, 那么其值会趋近于  $1 - \frac{1}{e} \approx$

0.623。在 Boosting 方法中,我们通过更加关注之前错误分类的记录,使用迭代过程自适应地改变训练数据的分布。一开始,所有的记录都被分配相同的权值。但是,不像 Bagging 方法,在每一轮的提升中权值是可以变化的:被错误分类的记录权值将会增加,同时正确分配的记录的权值将会降低。Boosting 方法的例子是 AdaBoost 算法。

分类器集成使用的例子在推荐领域里面非常实用。事实上,任何一个混合技术[16]都可以理解成以一种方式集成或另外几个分类器的集成。Tiemann 和 Pauws 的音乐推荐系统就是一个明显的实例,他们用集成学习方法来结合基于社交的和基于内容的推荐系统[70]。

实验结果显示,集成器能产生比其他任何孤立的分类器更好的结果。例如, Bell 等[11]在解决 Netflix 挑战、赢得大奖的方案中使用结合了 107 种不同的方法。他们的发现显示,本质上不同的方法比提升一种单一特殊技术的回报要好。为了从集合器中混合结果,他们采用线性回归方法。为了给每一个分类器生成权值,他们把测试数据集分成 15 个不同的部分,并且为每一个备份生成唯一的系数。在 Netflix 环境中的不同的集成方法可以追溯到其他的方法,如 Schclar 等的[67]或 Toescher 等的[71]。

自举方法也已经在推荐系统使用,例如, Freund 等提出一个被称为 RankBoost 的算法来结合用户的偏好[32]。他们应用这个算法在 CF 环境中来产生电影推荐。

### 2.3.8 评估分类器

推荐系统中被接受最常用的指标是预测兴趣(评分)和测量值的均方差(MAE)或均方根误差(RMSE)。这些指标在计算精度时对推荐系统的目标没有任何假设。但是,正如 McNee 等[51]指出的那样,除了精确度,还有许多指标来决定物品是否要被推荐。Herlocker 等[42]发表了推荐系统算法指标方法的综述。他们建议某些指标对于某些推荐任务可能更加合适。但是,如果在一类推荐算法和单个数据集上要根据经验来评估不同方法,他们不能验证这些指标。

下一步要考虑的是,“现实”中推荐系统的目的是产生一个 top-N 推荐列表,以及依赖于能多好地分辨出值得推荐的物品来评估这个推荐系统。如果把推荐看作分类问题,就可以使用评估分类器的著名指标,如准确度和召回率。在如下的段落中,我们将概述一部分这些指标及其在推荐评价中的应用。但是值得注意的是,学习算法和分类能够被多个准则来评估。这包含执行分类的准确率、训练数据的计算复杂度、分类的复杂度、噪声数据的敏感度、可扩展性等。但是在本章中我们将只关注分类性能。

为了评估一个模型,我们一般考虑以下的指标:真正(TP):分到类 A 且真的属于类 A 的实例数量;真负(TN):没有分到类 A 且真的不属于类 A 的实例数量;假正(FP):分到类 A 但不属于类 A 的实例数量;假负(FN):没有分到类 A 但属于类 A 的实例数量。

最常用来衡量模型性能是定义正确分类的(属于或不属于给定的类)实例和总的实例数量之间的比率:精确度 =  $(TP + TN) / (TP + TN + FP + FN)$ 。但是,精确度在许多的例子中有误导。想象一个带有 99 900 个类 A 的样本和 100 个类 B 的样本的两类分类问题。如果分类器简单地预测一切属于类 A,计算精度可能是 99.9%,但是模型性能值得怀疑,因为它从没有发现类 B 中的样本。改进这种估值的一种方法是定义代价矩阵,定义将类 B 的样本分给类 A 的代价。在真实的应用环境中,不同类型的错误可能的确有不同的代价。例如,如果 100 个样本对应一个组装线上有缺陷的飞机部分,不正确地拒绝一个没有缺陷的部分(1/99 900 的样本)相比于错误地把缺陷的部分当作好的,

这个代价是微不足道的。

模型性能的其他常用指标，特别是在信息检索中，是准确率和召回率。准确率，定义为  $P = TP / (TP + FP)$ ，是一种在分样本到类 A 中犯错误的指标。另一方面，召回率， $R = TP / (TP + FN)$ ，衡量没有留下本应该划分到类中的样本的程度。注意在大部分的例子中，当我们单独使用这两种指标时是有误导的。通过不分给任何的样本到类 A 可以建立有完美预测准确性的分类器（因此  $TP$  为零但  $FP$  也为零）。相反，通过分配所有的样本到类 A 中可以建立完美召回率的分类器。事实上，有一种结合了预测和召回率到一个单一指标中的指标：
$$F_1 = \frac{2RP}{R+P} = \frac{2TP}{2TP+FN+FP}。$$

有时我们会比较几个相互竞争的模型，而不是单独评估它们的性能。为此，我们使用在 20 世纪 50 年代开发的用来分析噪声信号的技术：接受者操作特征曲线（ROC）。ROC 曲线描述了正确击中和假警告之间的特征。每一个分类的性能用曲线上的点表示（见图 2.7）。

Ziegler 等[80]表示通过 top-N 列表指标评估推荐算法不能直接映射出用户的效率函数。但是，它的确解决了一些普遍接受的精确指标的限制，如 MAE。例如，Basu 等[10]，通过分析在评价规模中前四分之一被预测的物品哪些确实被用户评价为前四分之一。McLaughlin 和 Herlocker[50]提出一种修改后的精确指标，认为没有评价的物品计为不推荐。这个预测指标事实上代表了真实精确性的下限。尽管能够从准确率和召回率上直接得出 F-测量法，但是在推荐系统评估中很少有人用到。Huang 等[43]和 Bozzonnet 等[13]，以及 Miyahara 和 Pazzani[52]是使用这些指标的少数几个例子。

ROC 曲线也已经在评估推荐系统时使用。当在受到攻击下比较不同算法的性能时，Zhang 等[64]使用 ROC 曲线下的面积作为评估的指标。Banerjee 和 Ramanathan[8]也使用 ROC 曲线来比较不同模型的性能。

必须指出的是，好的评估指标的选择，即使在 top-N 推荐系统中，仍是一个研究点。许多作者提出了只用间接相关到这些传统的评估模式的指标。例如，Deshpande 和 Karypis[25]提出了命中率和平均逆命中率的使用。另一方面，Breese 等[15]将排序列表中推荐结果的效用指标定义成中立投票的函数。

注意，第 8 章会详细描述在推荐系统内容中这些评估指标的使用，因此如果你对这个问题感兴趣，你可以从那一章继续学习。

## 2.4 聚类分析

扩展 CF 分类器的最大问题是计算距离时的操作量，如发现最好的 K 近邻。如我们在 2.2.3 节中所看到那样，一种可能的解决方法是降维。但是，即使降低了特征维度，仍有

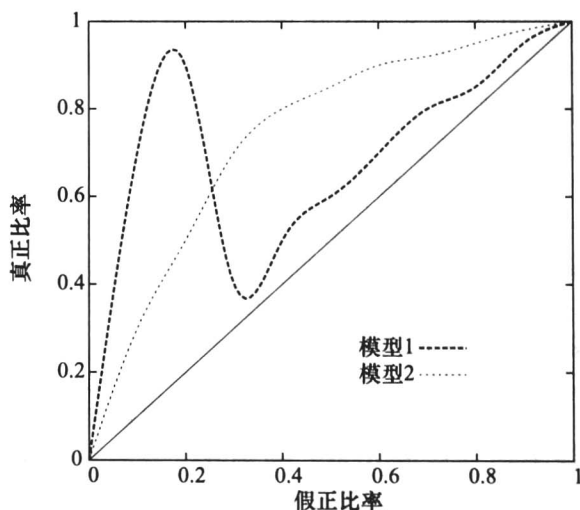


图 2.7 ROC 曲线的例子。模式 1 表现好于低假正比率，同时模式 2 是整体相当的一致，并且模式的假正比率优于模式 2 的 2.5

许多对象要计算距离,这就是聚类算法的用武之地。基于内容的推荐系统也是这样,检索相似对象也需要计算距离。由于操作量的减少,聚类可以提高效率。但是,不像降维方法,它不太可能提高精确度。因此,在设计推荐系统时必须谨慎使用聚类,必须小心地衡量提高效率和降低精度之间的平衡。

聚类[41],也称为无监督的学习,分配物品到一个组中使得在同一组中的物品比不同组中的物品更加类似:目的是发现存在数据中的自然(或者说是有意义)的组。相似度是由距离衡量决定的,如在2.2.1节中叙述的。聚类算法的目标是在最小化群内距离的同时最大化群间距离。

聚类算法有两个主要的类别:分层和划分。划分聚类算法把数据划分成非重合的聚类,使得每一个数据项确切在一个聚类中。分层聚类算法在已知聚类上继续聚合物品,生成聚类的嵌套集合,组成一个层级树。

许多聚类算法试图最小化一个函数来衡量聚类的质量。这样的质量函数一般被称为目标函数,因此聚类可以看作最优化的问题:理想聚类算法考虑所有可能数据划分,并且输出最小化质量函数的划分。但相应的最优化问题是NP难问题,因此许多算法采用启发式方法(例如,k-means算法中局部最优化过程最可能结束于局部最小)。主要问题还是聚类问题太难了,很多情况下要想找到最优解就是不可能的。同样的原因,特殊聚类算法的选择和它的参数(如相似度测量)取决于许多的因素,包括数据的特征。在下面的章节将描述k-means聚类算法和其他的候选算法。

#### 2.4.1 k-means

k-means聚类是一种分块方法。函数划分 $N$ 个物品的数据集到 $k$ 个不相关的子集 $S_j$ ,其中包含 $N_j$ 物品,以便于它们按照给定的距离指标尽可能地靠近。在分块中每一个聚类通过它的 $N_j$ 个成员和它的中心点 $\lambda_j$ 来定义。每一个聚类的中心点是聚类中所有其他物品到它的距离之和最小的那个点。因此,我们定义k-means算法作为迭代来最小化 $1 - \frac{1}{e} \approx 0.623$ ,其中 $x_n$ 是向量,代表第 $n$ 个物品, $\lambda_j$ 是在 $S_j$ 中物品的中心点,并且 $d$ 是距离尺度。k-means算法移动聚类间的物品直到 $E$ 不再进一步降低。

算法一开始会随机选择 $k$ 个中心点。所有物品都会被分配到它们最靠近的中心节点的类中。由于聚类新添加或是移出物品,新聚类的中心节点需要更新,聚类的成员关系也需要更新。这个操作会持续下去,直到再没有物品改变它们的聚类成员关系。算法第一次迭代时,大部分的聚类的最终位置就会发生,因此,跳出迭代的条件一般改变成“直到相对少的点改变聚类”来提高效率。

基础的k-means是极其简单和有效的算法。但是,它有几个缺陷:1)为了选择合适的 $k$ 值,假定有先验的数据知识;2)最终的聚类对于初始的中心点非常敏感;3)它会产生空聚类。k-means也有几个关于数据的缺陷:当聚类是不同的大小、密度、非球状形状时,就会有问题,并且当数据包含异常值时它也会有问题。

Xue等[77]提出一种在推荐环境中典型的聚类用法,通过使用k-means算法作为预处理步骤来帮助构造邻居。他们没有将邻居限制在用户所属的聚类内,相反是使用从用户到不同聚类中心点的距离作为预选阶段发现邻居。他们实现了基于聚类平滑技术,其技术是对于用户在聚类中的缺失值被典型聚类取代。他们的方法据称比标准的基于kNN的CF效果要好。相类似,Sarwar等[26]描述了一个方法来实现可扩展的kNN分



类器。他们通过平分 k-means 算法来划分用户空间,然后用这些聚类作为邻居的形成的基础。据称与标准的 kNN 的 CF 相比准确率降低了大约 5%。但是,他们的方法显著地提高了效率。

Connor 和 Herlocker[21]提出不同的方法,他们聚类物品而不是用户。使用 Pearson 相关相似度指标,他们尝试四种不同算法:平均链接分层聚集[39],对于分类属性的健壮聚类算法(ROCK)[40]:kMetis 和 hMetis<sup>⊖</sup>。尽管聚类的确提高了效率,但是所有的聚类技术的确比非分类基线精确度和覆盖度要差。最后,Li 等[60]以及 Ungar 和 Foster[72]提出一种非常类似的方法,使用 k-means 聚类来解决推荐问题的概率模型解释。

## 2.4.2 改进的 k-means

基于密度的聚类算法,诸如,DBSCAN 通过建立密度定义作为在一定范围内的点的数量。例如,DBSCAN 定义了三种点:核心点是在给定距离内拥有超过一定数量邻居的点;边界点没有超过指定数量的邻居但属于核心点邻居;噪声点既不是核心点也不是边界点。算法迭代移除掉噪声数据并且在剩下的点上进行聚类。

消息传递聚类算法是最近基于图聚类方法的系列之一。消息传递算法没有一开始就将节点的初始子集作为中心点,然后逐渐调适,而是一开始就将所有节点都看作中心点,一般称为标本。在算法执行时,这些点,现在已经是网络中的节点了,会交换消息直到聚类逐渐出现。相似传播是这种系列算法的代表,通过定义节点之间的两种信息来起作用:“责任”,反映了在考虑其他潜在标本的情况下,接收点有多适合作为发送点的标本;“可用性”,从候选标本发送到节点,它反映了在考虑其他选择相同标本的节点支持的情况下,这个节点选择候选标本作为其标本的合适程度。相似传播已经被应用到 DNA 序列聚类,在图形中人脸聚类,或者是文本摘要等不同问题,并且效果很好。

最后,分层聚类按照层级树(树枝形结构联系图)的结构产生一系列嵌套聚类。分层聚类不会预先假设聚类的既定数量。同样,任何数量的聚类都能够通过选择合适等级的树来获得。分层聚类有时也与有意义的分类学相关。传统的分层算法使用一个相似度或者距离矩阵来合并或分裂一个聚类。有两种主要方法来分层聚类。在聚集分层聚类中,我们以点作为个体聚类,并且每一个步合并最近的聚类对,直到只有一个(或是  $k$  个聚类)聚类剩下。分裂分层聚类从一个包含所有物品的聚类开始,并且每一个分裂每一聚类,直到每一聚类包含一个点(或是有  $k$  个聚类)。

就我们所知,诸如前面提到 k-means 的替代方法没有应用在推荐系统中。k-means 算法的简单和效率优于它的替代算法。基于密度或者是分层聚类方法在推荐系统领域能起的作用还不是很清楚。另一方面,消息传递算法已经显示了其高效的特点,并且基于图的范例很容易转换成推荐问题。在未来一段时间内我们看到这些算法的应用是可能的。

## 2.5 关联规则挖掘

关联规则挖掘关注于规则的发现,其他能够根据事务中出现其他物品来预测出现某个物品。两个物品被发现相关只意味着共同出现,但是没有因果关系。注意不要将这种技术与在 2.3.3 节中提到的基于规则的分类混淆。

⊖ <http://www.cs.umn.edu/karypis/metis>

我们定义物品集为一个或多个物品的集合(例如, (牛奶, 啤酒, 尿布))。 $k$ -物品集是包含  $k$  个物品的集合。给定物品的频繁度称为支持量(比如, (牛奶, 啤酒, 尿布)=131)。并且物品集的支持度是包含它的事务的比例(例如, (牛奶, 啤酒, 尿布)=0.12)。频繁物品集是支持度大于或等于最小支持度阈值的物品集。关联规则是公式  $X \Rightarrow Y$  的表达式, 其中  $X$  和  $Y$  是物品集。(例如, 牛奶, 尿布  $\Rightarrow$  啤酒)。在这个案例中, 关联规则的支持度是同时拥有  $X$  和  $Y$  的事务的比例。另一方面, 规则的置信度是  $Y$  中的物品有多经常出现在包含  $X$  的事务中。

给定一组事务集合  $T$ , 关联规则挖掘的目标是发现具有支持度大于或等于最小支持度阈值以及置信度大于或等于最小置信度阈值的所有规则。暴力法将会列出所有可能的关联规则, 为每一个规则计算支持度和置信度, 然后删除不满足两个条件的规则。但是, 这样的计算开销太大。因此, 我们采用两步方法: 1) 产生了所有支持度大于或等于最小支持度的物品集(频繁项集生成); 2) 从每一频繁物品集中产生高置信规则(规则产生)。

有几个技术来优化频繁物品集的产生。在一个广泛的意义上, 它们可以分成: 尝试最小化候选集数量( $M$ ), 降低事务量( $N$ ), 降低比较量数量( $NM$ )。但是最常用的方法是使用先验规则来降低候选数量。这个原则表明如果物品集是频繁的, 那么所有的子集也是频繁的。支持度的衡量标准已经验证了这一点, 因为一个物品集的支持度永远不会超过其子集的支持度。Apriori 算法是这个规则实际的实现。

给定一个频繁集  $L$ , 产生规则时的目的是发现所有满足最小的置信度需求的非空子集。如果  $|L|=k$ , 那么有  $2^k-2$  条候选关联规则。因此, 在生成频繁物品集时, 需要找到高效的方法来生成规则。对于 Apriori 算法, 我们能通过合并规则结果中共用相同前缀的两个规则来产生候选规则。

关联规则在发现模式和推动个性化市场营销方面的显著效果闻名已久[2]。但是, 尽管这些方法和推荐系统的目标之间有明显的关联, 但是它们还是没有成为主流。主要原因是这种方法类似于基于物品的 CF 但缺少灵活性, 因为它需要事务这个明确的概念——事件共同出现在某个给定的会话中。在第3章中我们将举一些有意义的例子, 其中一些表明关联规则仍有潜力。

Mobasher 等[53]提出一种基于关联规则的个性化网页系统。他们的系统基于用户的导航模式, 从共同出现的浏览页面来识别关联规则。他们在精确度和覆盖率指标方面优于基于 kNN 的推荐系统。Smyth 等[68]提出给推荐系统使用关联规则的两种不同的研究案例。在第一种案例中, 为了生成较好的物品-物品相似度指标, 他们从用户属性中使用先验算法来抽离物品关联规则。在第二种案例中, 他们应用关联规则到会话推荐中。这里的目标是发现共同发生的评论, 比如, 用户通过一个推荐物品的特定特征表明偏好。Lin 等[49]提出一种新的关联规则挖掘算法, 为了获得一个合适的有意义规则数量, 在挖掘期间调整规则的最小支持度, 因此解决了先前像 Apriori 这样算法的某些缺陷。他们挖掘在用户之间和物品之间的关联规则。测量出的精确度优于基于相关度推荐的报告值, 并且接近于更精巧的方法, 如 SVD 和 ANN 的结合。

最后, 如在 2.3.2 节中提到的那样, Cho 等[18]在一个网页商店推荐系统中结合了决策树和关联规则挖掘。在他们的系统, 关联规则的导入是为了链接相关的物品集。然后通过连接用户偏好和关联规则来计算得出推荐结果。他们在不同的事务集中寻找关联规则, 如商品, 购物车, 点击率。他们用启发式学习给每一个事务集中规则附加权重。例如, 商品关联规则权重大于点击关联规则。

## 2.6 总结

本章介绍了在设计推荐系统中可能用到的主要的数据挖掘方法和技术。我们也总结了在文献中提到的用法,提供了如何以及在哪用到它们一些粗略指导。

我们从综述在预处理步骤可能用到的技术开始。首先,2.2.1节回顾了如何选择合适的距离衡量指标。在后面的步骤中大部分的方法需要它。余弦相似度和皮尔逊相关度是一般可接受最好的选择。尽管付出了许多的努力来提高这些距离指标,但是最近的工作似乎表明距离函数的选择不是这么重要。然后,在2.2.2节回顾了最基础的抽样,其应用是为了选择原始大数据集的子集,或者是划分训练和测试集。最后,我们讨论降维技术的使用,诸如,在2.2.3节中主成分分析(PCA)和奇异值分解(SVD)作为一种方法来解决维度灾难问题。我们解释了一些使用降维技术的成功案例,特别是在Netflix大奖的环境中。

在2.3节中,我们回顾了主要的分类方法:即近邻,决策树,基于规则分类,贝叶斯网络,人工神经网络,支持向量机。我们看到,尽管kNN(见2.3.1节)的CF是首选的方法,但是所有这些方法都可以应用在不同的环境中。决策树(见2.3.2节)可以被用来导出基于物品内容的模型或者是模拟系统的特殊部分。决策规则(见2.3.3节)可以从预先存在的决策树中推导出,或者是被用来引入商业或者是领域知识。贝叶斯网络(见2.3.4节)是基于内容的推荐中一个流行的方法,但它也可以用来生成一个基于模型的协同过滤系统。类似的方法,人工神经网络能够被用来导出基于模型的推荐,也可以用来结合/混合其他几种算法。最后,支持向量机(见2.3.6节)作为一种方法来推断出基于内容的分类或者是导出CF模型而流行。

对于推荐系统来说,选择合适的分类器不容易,尤其是一些感知判断任务和数据依赖的情况下。在CF的案例中,一些结果似乎表明基于模型方法,使用如SVM或者是贝叶斯网络,能够稍微提高标准kNN分类的性能。但是,这些结果不显著并且很难推广。在基于内容的推荐系统的例子中有些证据表明,在一些例子中贝叶斯网络执行效果比简单方法要好,如决策树。但是,更加复杂的非线性分类,如ANN或SVM,执行效果是否更好还不是很清楚。

因此,给特定的推荐任务选择合适分类器在今天仍有许多探索的地方。实际的经验规则是从最简单的方法做起,并且只有在性能的提升值得时才采用复杂方法。性能增益应该平衡不同的维度,如预测精确度或计算效率。

我们在2.4节中回顾了聚类算法。聚类在推荐系统中一般被用来提高性能。不管是在用户空间还是物品空间,较早进行聚类步骤都能减少随后要做的计算距离的操作数量。但是,这一般以较低的精确度为代价,所以处理时要慎重。事实上,通过使用降维技术(如SVD)提高效率在一般的例子中是好的选择。与分类相反,没有那么多的聚类算法在推荐系统的环境中使用。k-means(见2.4.1节)算法由于简单和相对有效,很难找到实用的替代者。我们在2.4.2节中综述了它们中一些算法,如分层聚类或消息传递算法。尽管这些技术还没有应用在推荐系统中,但为将来的研究提供了有希望的出路。

最后,在2.5节中,我们描述了关联规则并总结了它们在推荐系统的使用。关联规则为推荐物品提供了直观的框架,只要有一个显式或隐式的事务。尽管存在有效的算法来计算关联规则,而且已经被证明比标准kNN的CF准确率好,但是他们仍不是受青睐的方法。

在设计推荐系统中选择正确的数据挖掘技术是一个复杂的任务,其一定受许多特殊问题约束。但是,我们希望本章中技术和经验的简短综述能够帮助读者做出更加合理的决定。除此之外,我们也发现了有待进一步提高的领域和令人兴奋的研究点,以及接下来有

待研究的相关研究点。

## 致谢

本章的完成受到来自 Generalitat de Catalunya 的 ICREA 部分基金支持。

## 参考文献

1. Adomavicius, G., and Tuzhilin, A., Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
2. Agrawal, R., and Srikant, R., Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, 1994.
3. Amatriain, X., Lathia, N., Pujol, J.M., Kwak, H., and Oliver, N., The wisdom of the few: A collaborative filtering approach based on expert opinions from the web. In *Proc. of SIGIR '09*, 2009.
4. Amatriain, X., Pujol, J.M., and Oliver, N., I like it... i like it not: Evaluating user ratings noise in recommender systems. In *UMAP '09*, 2009.
5. Amatriain, X., Pujol, J.M., Tintarev, N., and Oliver, N., Rate it again: Increasing recommendation accuracy by user re-rating. In *Recys '09*, 2009.
6. Anderson, M., Ball, M., Boley, H., Greene, S., Howse, N., Lemire, D., and S. McGrath. Racofi: A rule-applying collaborative filtering system. In *Proc. IEEE/WIC COLA '03*, 2003.
7. Baets, B.D., Growing decision trees in an ordinal setting. *International Journal of Intelligent Systems*, 2003.
8. Banerjee, S., and Ramanathan, K., Collaborative filtering on skewed datasets. In *Proc. of WWW '08*, 2008.
9. Basu, C., Hirsh, H., and Cohen, W., Recommendation as classification: Using social and content-based information in recommendation. In *In Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 714–720. AAAI Press, 1998.
10. Basu, C., Hirsh, H., and Cohen, W., Recommendation as classification: Using social and content-based information in recommendation. In *AAAI Workshop on Recommender Systems*, 1998.
11. Bell, R.M., Koren, Y., and Volinsky, C., The bellkor solution to the netflix prize. Technical report, AT&T Labs Research, 2007.
12. Bouza, A., Reif, G., Bernstein, A., and Gall, H., Semtree: ontology-based decision tree algorithm for recommender systems. In *International Semantic Web Conference*, 2008.
13. Bozzon, A., Prandi, G., Valenzise, G., and Tagliasacchi, M., A music recommendation system based on semantic audio segments similarity. In *Proceeding of Internet and Multimedia Systems and Applications - 2008*, 2008.
14. Brand, M., Fast online svd revisions for lightweight recommender systems. In *SIAM International Conference on Data Mining (SDM)*, 2003.
15. Breese, J., Heckerman, D., and Kadie, C., Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, page 4352, 1998.
16. Burke, R., Hybrid web recommender systems. pages 377–408. 2007.
17. Cheng, W., J. Hühn, and E. Hüllermeier. Decision tree and instance-based learning for label ranking. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 161–168, New York, NY, USA, 2009. ACM.
18. Cho, Y., Kim, J., and Kim, S., A personalized recommender system based on web usage mining and decision tree induction. *Expert Systems with Applications*, 2002.
19. Christakou, C., and Stafylopatis, A., A hybrid movie recommender system based on neural networks. In *ISDA '05: Proceedings of the 5th International Conference on Intelligent Systems Design and Applications*, pages 500–505, 2005.
20. Cohen, W., Fast effective rule induction. In *Machine Learning: Proceedings of the 12th International Conference*, 1995.
21. Connor, M., and Herlocker, J., Clustering items for collaborative filtering. In *SIGIR Workshop on Recommender Systems*, 2001.
22. Cover, T., and Hart, P., Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.

23. Cristianini, N., and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March 2000.
24. Deerwester, S., Dumais, S.T., Furnas, G.W., L. T. K., and Harshman, R., Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41, 1990.
25. Deshpande, M., and Karypis, G., Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.*, 22(1):143–177, 2004.
26. B. S. et al. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the Fifth International Conference on Computer and Information Technology*, 2002.
27. K. O. et al. Context-aware svm for context-dependent information recommendation. In *International Conference On Mobile Data Management*, 2006.
28. P. T. et al. *Introduction to Data Mining*. Addison Wesley, 2005.
29. S. G. et al. Tv content recommender system. In *AAAI/IAAI 2000*, 2000.
30. S. H. et al. Aimed- a personalized tv recommendation system. In *Interactive TV: a Shared Experience*, 2007.
31. T. B. et al. A trail based internet-domain recommender system using artificial neural networks. In *Proceedings of the Int. Conf. on Adaptive Hypermedia and Adaptive Web Based Systems*, 2002.
32. Freund, Y., Iyer, R., Schapire, R.E., and Singer, Y., An efficient boosting algorithm for combining preferences. *Mach. J. Learn. Res.*, 4:933–969, 2003.
33. Frey, B.J., and Dueck, D., Clustering by passing messages between data points. *Science*, 307, 2007.
34. Friedman, N., Geiger, D., and Goldszmidt, M., Bayesian network classifiers. *Mach. Learn.*, 29(2-3):131–163, 1997.
35. Funk, S., Netflix update: Try this at home, 2006.
36. Ghani, R., and Fano, A., Building recommender systems using a knowledge base of product semantics. In *2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*, 2002.
37. Goldberg, K., Roeder, T., Gupta, D., and Perkins, C., Eigentaste: A constant time collaborative filtering algorithm. *Journal Information Retrieval*, 4(2):133–151, July 2001.
38. Golub, G., and Reinsch, C., Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420, April 1970.
39. Gose, E., Johnsonbaugh, R., and Jost, S., *Pattern Recognition and Image Analysis*. Prentice Hall, 1996.
40. Guha, S., Rastogi, R., and Shim, K., Rock: a robust clustering algorithm for categorical attributes. In *Proc. of the 15th Intl Conf. On Data Eng.*, 1999.
41. Hartigan, J.A., *Clustering Algorithms (Probability & Mathematical Statistics)*. John Wiley & Sons Inc, 1975.
42. Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.T., Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.
43. Huang, Z., Zeng, D., and Chen, H., A link analysis approach to recommendation under sparse data. In *Proceedings of AMCIS 2004*, 2004.
44. Isaksson, A., Wallman, M., H. Göransson, and Gustafsson, M.G., Cross-validation and bootstrapping are unreliable in small sample classification. *Pattern Recognition Letters*, 29:1960–1965, 2008.
45. Jolliffe, I.T., *Principal Component Analysis*. Springer, 2002.
46. Kang, H., and Yoo, S., Svm and collaborative filtering-based prediction of user preference for digital fashion recommendation systems. *IEICE Transactions on Inf & Syst*, 2007.
47. Kurucz, M., Benczur, A.A., and Csalogany, K., Methods for large scale svd with missing values. In *Proceedings of KDD Cup and Workshop 2007*, 2007.
48. Lathia, N., Hailes, S., and Capra, L., The effect of correlation coefficients on communities of recommenders. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2000–2005, New York, NY, USA, 2008. ACM.
49. Lin, W., and Alvarez, S., Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery Journal*, 6(1), 2004.
50. M. R. McLaughlin and Herlocker, J.L., A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In *Proc. of SIGIR '04*, 2004.
51. S. M. McNee, Riedl, J., and Konstan, J.A., Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1097–1101, New York, NY, USA, 2006. ACM Press.
52. Miyahara, K., and Pazzani, M.J., Collaborative filtering with the simple bayesian classifier. In *Pacific Rim International Conference on Artificial Intelligence*, 2000.



53. Mobasher, B., Dai, H., Luo, T., and Nakagawa, M., Effective personalization based on association rule discovery from web usage data. In *Workshop On Web Information And Data Management, WIDM '01*, 2001.
54. Nikovski, D., and Kulev, V., Induction of compact decision trees for personalized recommendation. In *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pages 575–581, New York, NY, USA, 2006. ACM.
55. M. P. Omahony. Detecting noise in recommender system databases. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI06)*, 29th1st, pages 109–115. ACM Press, 2006.
56. Paterek, A., Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD Cup and Workshop 2007*, 2007.
57. Pazzani, M.J., and Billsus, D., Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.
58. Pronk, V., Verhaegh, W., Proidl, A., and Tiemann, M., Incorporating user control into recommender systems based on naive bayesian classification. In *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pages 73–80, 2007.
59. Pyle, D., *Data Preparation for Data Mining*. Morgan Kaufmann, second edition edition, 1999.
60. Li, B., K.Q., Clustering approach for hybrid recommender system. In *Web Intelligence 03*, 2003.
61. Quinlan, J.R., Induction of decision trees. *Machine Learning*, 1(1):81–106, March 1986.
62. Rendle, S., and L. Schmidt-Thieme. Online-updating regularized kernel matrix factorization models for large-scale recommender systems. In *Recsys '08: Proceedings of the 2008 ACM conference on Recommender Systems*, 2008.
63. Rokach, L., Maimon, O., *Data Mining with Decision Trees: Theory and Applications*, World Scientific Publishing (2008).
64. Zhang, J., F.S., Ouyang, Y., and Makedon, F., Analysis of a low-dimensional linear model under recommendation attacks. In *Proc. of SIGIR '06*, 2006.
65. Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., Incremental svd-based algorithms for highly scalable recommender systems. In *5th International Conference on Computer and Information Technology (ICCIT)*, 2002.
66. Sarwar, B.M., Karypis, G., Konstan, J.A., and Riedl, J.T., Application of dimensionality reduction in recommender systems a case study. In *ACM WebKDD Workshop*, 2000.
67. Schlar, A., Tsikinovsky, A., Rokach, L., Meisels, A., and Antwarg, L., Ensemble methods for improving the performance of neighborhood-based collaborative filtering. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 261–264, New York, NY, USA, 2009. ACM.
68. Smyth, B., K. McCarthy, Reilly, J., D. O'Sullivan, L. McGinty, and Wilson, D., Case studies in association rule mining for recommender systems. In *Proc. of International Conference on Artificial Intelligence (ICAI '05)*, 2005.
69. Spertus, E., Sahami, M., and Buyukkokten, O., Evaluating similarity measures: A large-scale study in the orkut social network. In *Proceedings of the 2005 International Conference on Knowledge Discovery and Data Mining (KDD-05)*, 2005.
70. Tiemann, M., and Pauws, S., Towards ensemble learning for hybrid music recommendation. In *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pages 177–178, New York, NY, USA, 2007. ACM.
71. Toeschler, A., Jahrer, M., and Legenstein, R., Improved neighborhood-based algorithms for large-scale recommender systems. In *KDD-Cup and Workshop 08*, 2008.
72. Ungar, L.H., and Foster, D.P., Clustering methods for collaborative filtering. In *Proceedings of the Workshop on Recommendation Systems*, 2000.
73. Witten, I.H., and Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, second edition edition, 2005.
74. Wu, M., Collaborative filtering via ensembles of matrix factorizations. In *Proceedings of KDD Cup and Workshop 2007*, 2007.
75. Xia, Z., Dong, Y., and Xing, G., Support vector machines for collaborative filtering. In *ACM-SE 44: Proceedings of the 44th annual Southeast regional conference*, pages 169–174, New York, NY, USA, 2006. ACM.
76. Xu, J., and Araki, K., A svm-based personal recommendation system for tv programs. In *Multi-Media Modelling Conference Proceedings*, 2006.

77. Xue, G., Lin, R., Yang, C., Xi, Q., Zeng, W., H.-, Yu, J., and Chen, Z., Scalable collaborative filtering using cluster-based smoothing. In *Proceedings of the 2005 SIGIR*, 2005.
78. Yu, K., Tresp, V., and Yu, S., A nonparametric hierarchical bayesian framework for information filtering. In *SIGIR '04*, 2004.
79. Zhang, Y., and Koren, J., Efficient bayesian hierarchical user modeling for recommendation system. In *SIGIR 07*, 2007.
80. Ziegler, C., McNeer, S. M., Konstan, J.A., and Lausen, G., Improving recommendation lists through topic diversification. In *Proc. of WWW '05*, 2005.
81. Zurada, J., *Introduction to artificial neural systems*. West Publishing Co., St. Paul, MN, USA, 1992.

## 基于内容的推荐系统：前沿和趋势

Pasquale Lops、Marco de Gemmis 和 Giovanni Semeraro

**摘要** 推荐系统以个性化的方式指引用户在众多选择中找到感兴趣的东西。基于内容的推荐系统会发掘用户曾经喜欢的产品，从而尝试去推荐类似的产品使其满意。事实上，基于内容的推荐系统主要的处理方式在于利用用户已知的偏好、兴趣等属性和物品内容的属性相匹配，以此为用户推荐新的感兴趣的物品。本章概述了各种基于内容的推荐系统，目的是在其各种设计原理和实现方法中理出头绪。本章第一部分介绍了基于内容推荐的基本概念、专业术语、高层次的体系结构和主要的优缺点。第二部分通过详尽描述能够表示物品和用户信息的经典先进技术，给出了几个应用领域内使用的最先进技术的概述。同时，也阐述了一些被广泛使用的学习用户兴趣的技术。最后一部分讨论了推荐系统的趋势和下一代推荐系统的研究方向，其中描述了考虑到在词汇表不断演变情况下用户产生内容(UGC)的作用，以及为用户提供一些偶然性的推荐的挑战，即推荐出乎意料地能够让用户感兴趣，而又无法通过其他方法发现的物品。

### 3.1 简介

网络上和数字图书馆中，存在着大量而丰富的信息，由于它们的动态性和多样性，很难快速找出我们想要的或最能满足我们需求的东西。

因此，用户建模和个人资料访问的作用变得越来越重要：根据喜好和品位，用户需要个性化的支持来从大量信息中筛选出可用信息。

大量的信息来源显示，推荐系统是能够满足用户个性化需求的一种方式[73]。推荐系统在巨大的可能选择范围内引导用户发现感兴趣的或有用的个性化推荐结果[17]。推荐算法把用户的兴趣作为输入来产生一个推荐列表。亚马逊的推荐算法用于为每个用户定制一个网上商店。例如，为软件工程师显示有关编程的主题，为新妈妈显示的是婴儿玩具[50]。

物品推荐的问题已经有广泛的研究，现有两类主要方式。基于内容的推荐系统试图推荐给定用户过去喜欢的相似物品，而协同过滤推荐方式的系统识别出拥有相同喜好的用户，并推荐他们喜欢过的物品[7]。

本章中，我们对基于内容的推荐系统进行了全面而系统的研究，其目的有两方面：

- 提供研究现状的概览，着重讨论最有效的技术和它们被采纳的应用领域。
- 指明能够引领下一代基于内容的推荐系统的未来研究趋势和方向。

---

Pasquale Lops, Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona, 4, Bari (Italy) e-mail: lops@di.uniba.it

Marco de Gemmis, Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona, 4, Bari (Italy) e-mail: degemmis@di.uniba.it

Giovanni Semeraro, Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona, 4, Bari (Italy) e-mail: semeraro@di.uniba.it

翻译：吴涛，吉林大学-丁彬钊，徐世超 审核：胡聪(胡户主)，北京大学-王雪丽，郑州大学-吴宾

本章结构如下：3.2 节介绍基于内容的推荐系统的基本概念和术语。通过讲解经典的基于内容的推荐系统架构来理解架构中的主要部件，以及在产生推荐结果的过程中，这种类型的推荐技术的优缺点。3.3 节全面介绍最前沿的基于内容的推荐系统，还全面描述了如何表示物品和用户信息的经典方法和最新技术，以及用户信息的学习方法。3.4 节阐述了基于内容的推荐系统技术的趋势和未来研究方向。3.5 节进行了总结。

3.2 基于内容的推荐系统的基础

基于内容的推荐系统通过分析一系列用户之前已评分物品的文档和(或)描述，从而基于用户已评分对象的特征建立模型或个人信息[63]。个人信息是用户兴趣的结构化描述，并且被应用在推荐新的感兴趣的物品中。推荐的主要处理过程是将用户个人信息的特征和内容对象的特征相匹配，结果就是用户对某个对象感兴趣程度的评价。如果这份个人信息准确地反映了用户的偏好，信息访问处理过程则能带来巨大的优势。例如，通过判断这个用户是否对指定网页感兴趣，以及在不感兴趣的情况下阻止显示，从而达到有效过滤搜索结果的效果。

3.2.1 基于内容的推荐系统的高层次结构

基于内容的信息过滤系统需要描述物品和生成用户个人信息的恰当技术，以及一些能够比较用户个人信息和物品描述的策略。一个基于内容的推荐系统的高层次结构如图 3.1 所示。推荐的过程有三个阶段，每一阶段都由独立的部件控制。

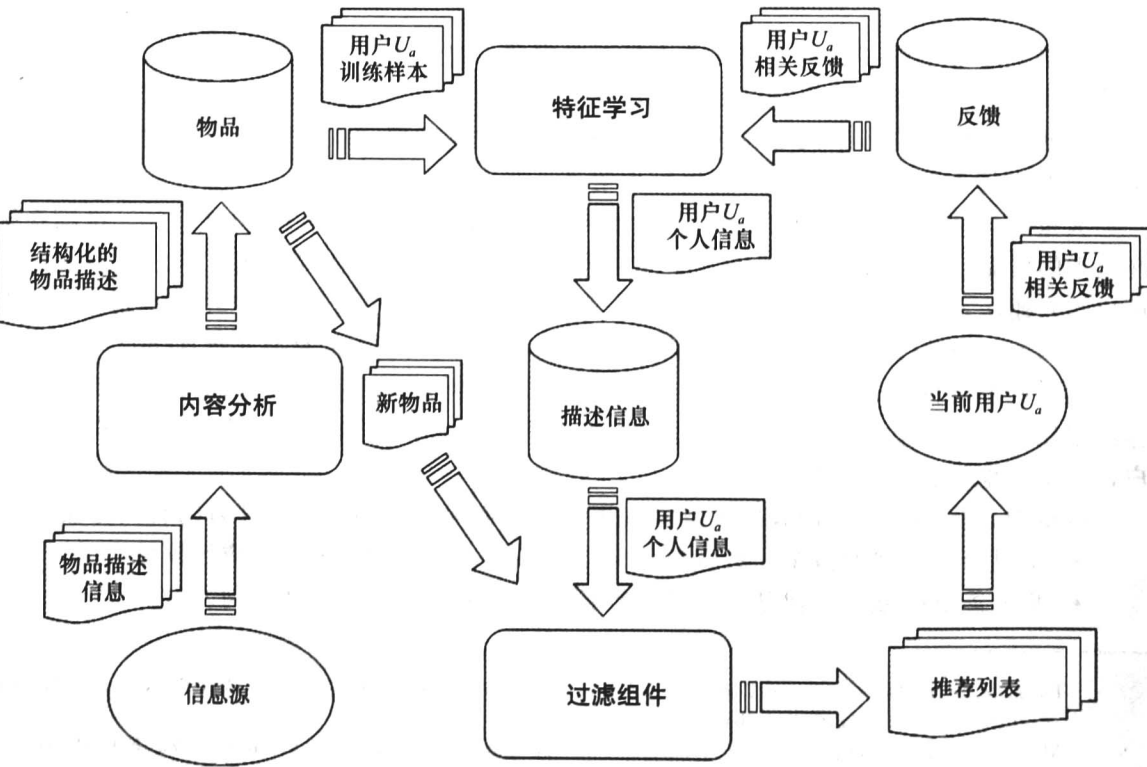


图 3.1 基于内容的推荐系统的高层次结构

- **内容分析器**：当信息没有结构化时(如文本)，某些预处理阶段需要抽取相关的结构化信息。这个部件的主要功能就是将来自信息源的对象(如文档、网页、新闻、产品描述等)的内容表示成恰当的格式，以便于下一阶段的处理。数据对象经过特征抽取技术的分析，目的是将原始信息空间转换到想要的物品描述格式(如将网页表示成关键词向量)。这样的描述格式作为信息学习器和过滤组件的输入。
- **信息学习器**：这个模块收集了有关用户偏好的数据特征，并试图去泛化这些数据，从而构建用户特征信息。泛化策略通常是通过机器学习技术实现的[61]，它可以从用户过去喜欢的或不喜欢的物品中推断出一个用户的兴趣模型。例如，网页推荐的信息学习器可以实现相关的反馈方法[75]，通过学习技术将正负样例向量组合到一个表示用户特征的模型向量中。训练样本是由用户提供的具有正负反馈的网页。
- **过滤组件**：这个模块将用户个人信息和物品在表示空间进行匹配，利用用户个人信息来推荐相关物品。这个组件的结果是一个二元的或者连续型的相关性判断(使用某种相似度来计算[42])，后者能生成一个潜在感兴趣物品的排名清单。在上面提过的例子中，这种匹配是通过计算原型向量和物品向量的余弦相似度得到的。

推荐步骤的第一个阶段是由内容分析器完成的，它通常是借鉴了信息检索系统的技术[80, 6]。来自信息源的物品描述经过内容分析器，从非结构化的文本中抽取特征(如关键词、n-grams、概念等)，从而得到结构化的物品描述，并储存在被表示物品库中。

为了结构化和更新活跃用户  $u_a$  (必须为其提供推荐的用户) 的个人信息，该用户对物品的偏好反应是通过某些渠道收集并记录在反馈库中的。这些被称作注释[39]或反馈的相互作用和物品的相关描述一起被用在模型学习的过程中，这些信息对实际中预测新的相关物品的表示非常有用。因此，即使没有提供任何反馈，用户也可以清晰地定义他们自己感兴趣的领域作为初始的个人信息。

通常情况下，我们能够区分这两种类型的相关性反馈：正面的信息(用户喜欢的特征)和负面的信息(用户不感兴趣的特征[43])。

两种不同的技术都能用来记录用户的反馈。当系统要求用户明确评价物品时，这项技术通常称作“显式反馈”；反之，则称作“隐式反馈”，由于反馈来自监控和分析用户的行为，所以它不需要任何活跃用户的参与。

确切的评价能够表明用户对一个物品相关或感兴趣的程度[74]。主要有三种方式得到显式的相关性反馈：

- **喜欢/不喜欢**：利用一个简单的二元化评分刻度，将物品分成“相关的”或“不相关的”两大类，如[12]。
- **评分**：经常用来评价物品的一个离散的数值刻度，详情参见文献[86]。当然，标记化的评价也可以映射到数值刻度，如在 Syskill & Webert[70]中一样，把用户对网页的评价划分为热门、一般、冷门。
- **文本评论**：收集并展示单一物品的评论给用户，使其成为用户加快决策过程的一种方式，详情参见文献[72]。例如，在 Amazon 或 eBay 上，用户的反馈可以帮助其他用户判断一件物品是否被大众所接受。文本评论是有益的，但是这些评论也会对用户造成负担，因为她必须阅读和理解每条评论，并决定哪些评论是正面的哪些是负面的，以及这些评论的程度。文献[71]从情感计算研究领域中提出的先进技术，使得基于内容的推荐系统能够自动执行这种分析。



即使数值/符号刻度的可采纳性会增加用户的认知负荷，而且也许不能完全得到用户对物品的评价，显式反馈仍然具有简单的优点。隐式反馈的方法是基于对用户在某样物品上的特定行为进行相关性评分赋值，如保存、删除、印刷、收藏等。这个方法的主要优点是不需要用户的直接参与，当然偏差是可能发生的，如在阅读的时候被电话打断。

为了建立活跃用户  $u_a$  的个人信息，必须定义用户  $u_a$  的训练集  $TR_a$ 。 $TR_a$  是一个成对的  $\langle I_k, r_k \rangle$  的集合，其中  $r_k$  是用户对物品描述  $I_k$  的评分。给定一组有评分的物品描述，信息学习器通过监督式学习算法生成一个预测模型——用户信息，用户信息通常存储在信息资源库里，并被之后的过滤组件所使用。给定一个新的物品描述，通过比较存储在用户信息资源库里的用户偏好和表述物品的特征，过滤组件将会预测活跃用户是否对它感兴趣。过滤组件通常会实现根据用户个人信息相关性的程度对潜在感兴趣物品进行排序的策略。排名靠前的物品会形成一个推荐列表  $L_a$ ，并被推荐给用户  $u_a$ 。用户的品味通常时刻改变，因此必须维护最新的变更，并提供给信息学习器去自动更新用户个人信息。通过让用户表明对  $L_a$  中的物品满意与否，能从生成的推荐中收集到更深层次的反馈。在收集了这些反馈后，学习过程将在新的训练数据集上再次执行，并将学习的结果应用于生成用户的最新兴趣。随着时间的推移，“反馈—学习”的循环迭代使得该系统可以考虑到用户偏好的动态变化。

### 3.2.2 基于内容过滤的优缺点

基于内容的推荐与基于协同过滤的推荐相比有以下优点：

- **用户独立性：**基于内容的推荐仅使用当前用户提供的评分来构建自己的个人信息。而协同过滤的方法需要其他用户的评分，来发现该用户最近的近邻，例如，由于对相同的物品评分相似而品味相似的用户。这时，只有当前用户最近邻很喜欢的物品才有可能推荐给当前用户。
- **透明度：**通过显式地列出使得物品出现在推荐列表中的内容特征或描述，可以解释推荐系统是如何工作的。这些物品特征是决定是否信任该推荐的指标。相反，协同过滤系统是一个黑盒子，对一个推荐物品的唯一解释是相似品味的未知用户喜欢过该物品。
- **新物品：**基于内容的推荐系统在没有任何用户评分的情况下也可以进行推荐。因此，新物品没有第一次评分会影响协同过滤推荐系统，因为协同过滤推荐系统仅依赖于用户的偏好产生推荐。所以只有当一个新物品被一系列用户评分之后，系统才可能推荐它。

尽管如此，基于内容的推荐系统也有以下一些缺点：

- **可分析的内容有限：**基于内容的推荐技术有一个天然的限制，即与推荐对象相关的特征数量和类型上的限制，不管是自动还是手动的。领域知识一般是必需的，例如，对于电影推荐，系统需要知道电影的演员、导演，有时候领域本体也是需要的。当分析的物品内容信息不足以区分哪些物品是用户喜欢的、哪些物品是用户不喜欢的时候，没有任何基于内容的推荐系统可以给出合适的推荐。有些解释只能获取物品内容的某些方面，但是还有很多别的方面也能影响用户体验。举个例子，在玩笑或者诗词里，没有足够的词频信息去为用户兴趣建模，这时，情感计算的技术就会更适用。此外，对于网页来说，文本特征抽取技术完全忽略其美学特征和附加的多媒体信息。

总之，不论是手动还是自动为物品分配特征，都不足以定义物品不同的特点，而这些特点被证明对提取出用户兴趣是必要的。

- **过度特化**：基于内容的推荐在本质上无法发现一些出人意料物品。系统建议的物品和用户的个人信息高度匹配的时候，给用户的推荐也将会是与已有的评分物品相似的物品。这个缺点主要是由于基于内容的系统产生的推荐物品在新颖性上的缺陷，称作惊喜度问题。举例来说，当一个用户只评价了 Stanley Kubrick 导演的电影，那么她得到的推荐就只有这种类型的电影。一个“完美”的基于内容的技术可能很少发现任何新颖的东西，这限制了使用它的应用程序的范围。
- **新用户**：在一个基于内容的推荐系统可以真正理解用户偏好且给出准确的推荐之前，需要收集足够的评分。因此，当只有很少的评分可用的时候，即对于新用户来说，系统不能提供可靠的推荐。

接下来，将就采用何种策略来处理对上面提出的问题，进行介绍和讨论。更具体地，会阐述利用常识和特定领域的知识来提高内容解释的新技术(3.3.1.3节~3.3.1.4节)。通过提供新的特征可能有助于克服传统的内容分析方式的限制，如 WordNet[60, 32]或 Wikipedia 概念，帮助物品用一种更准确透明的方式进行推荐。此外，将推荐过程中用户定义词典，如大众分类，作为扩展词表加入考虑并进行整合的过程，将在 3.4.1 节进行介绍。

使用惊喜度推荐，即新颖性很高的用户感兴趣的物品，来满足用户的可能方式将作为解决过度特化问题的解决方案进行分析(3.4.2节)。

最后，将会介绍克服新用户问题的各种不同策略。其中，当在特定用户的评分很少或者没有的情况下推荐时，社区用户所提供的社交标签可以作为推荐的反馈应用到系统中(3.4.1.1节)。

### 3.3 基于内容的推荐系统的现状

顾名思义，基于内容的推荐是利用物品的内容数据来预测它和用户个人信息的相关性。基于内容的推荐系统的研究涉及计算机科学的许多方面，尤其是在信息检索[6]和人工智能领域。

在信息检索领域，推荐技术研究的想象力来自将用户搜索推荐结果看作一个信息检索的过程。在信息检索系统中，用户需要给出一次性查询信息(经常是一个关键词列表)，而在信息过滤系统，用户的信息需求被表示成他的个人信息。由于用来描述物品的属性在数量和类型上的差异，待推荐物品也会有较大差异。每个物品当然可以用一组已知数值的少量相同的属性来表示，但是这种方法不是非常合适如 Web 网页、新闻、电子邮件或者文档等非结构化数据表示的对象。在这种属性值未明确定义的情况下，基于信息检索研究的文档模型技术更有用。

从人工智能的观点出发，推荐任务可以被映射成一个利用用户过去知识来学习的学习问题。最简单地，用户的个人信息形式为用户指定的关键词或准则，它反映了用户长期的兴趣。往往建议系统学习用户个人信息，而不是强迫用户自己提供。这就广泛应用了机器学习技术，其目标是基于过去已知并被用户显式或隐式地标记为是否感兴趣的信息，来学习分类新物品。给定这些已标记的物品信息，机器学习的方法能够生成一个预测模型，可以帮助判断目标用户对给定新物品是否感兴趣。

3.3.1 节阐述了多个物品信息表示技术，从传统的文本表示，到整合了本体和百科全书式知识等更先进的技术。3.3.2 节讨论了适合上面描述的表示方法的推荐算法。

### 3.3.1 物品表示

推荐给用户的物品可以表示成一系列特征,也称为属性(attribute或property)。例如,在一个电影推荐的应用中,描述一个电影的特征有演员、导演、类型、主题等。当每个物品由一系列相同的属性表示,并且知道这些属性所有可能的取值时,该物品就被表示成了结构化数据。在此情况下,许多机器学习的算法可以用来学习用户个人信息[69]。

在大多数基于内容过滤的系统,物品表示是从Web网页、电子邮件、新闻或产品描述中抽取的文本特征。不像结构化数据,它们没有明确定义的属性。由于自然语言的歧义,在学习用户个人信息时,文本特征带来大量混乱。问题是传统的基于关键词的个人信息不能获取用户兴趣的语义,因为他们主要是通过字符串匹配操作来进行的。如果一个字符串或者某些语法变量同时在个人信息和文档中被发现,那么匹配就成功,文档就被认为是相关的。字符串匹配有如下问题:

- 一词多义,一个词有多个释义。
- 同义,多个词有相同的意思。

导致的结果就是,由于同义词的存在,当个人信息不包含文档中的准确关键词时,相关的信息就会遗漏;同时由于一词多义,错误的文档也被认为是相关的。

语义分析及其与个性化模型的结合是文献中提到的最具有创新性和最有趣的解决问题的方法之一。主要的思路是采用知识库,如词典或本体,来注释物品和表示个人信息,由此获得一个用户信息所需的语义翻译。在3.3.1.1节,先回顾一个依赖于该模型的“传统”系统,之后将介绍基于本体论和世界知识的语义分析技术。

#### 3.3.1.1 基于关键字向量空间模型

大多数基于内容的推荐系统使用相对简单的检索模型,例如,关键字匹配或者基于TF-IDF权重的向量空间模型(Vector Space Model, VSM)。向量空间模型是一个文本文档的空间表示方法。在该模型中,每个文档被表示成一个 $n$ 维空间中的向量,每一维对应给定文档集合词汇表中的一个词。形式上,每篇文档被表示成词权重的向量,其中权重表示这篇文档和该词的关联度。让 $D=\{d_1, d_2, \dots, d_N\}$ 表示一个文档集合或语料库, $T=\{t_1, t_2, \dots, t_N\}$ 表示词典,即语料库中词的集合。词典 $T$ 从某些标准的自然语言处理操作中获取,如分词、停用词移除、变形[6]。每篇文档 $d_j$ 表示为 $n$ 维向量空间上的一个向量,从而 $d_j=\{w_{1j}, w_{2j}, \dots, w_{nj}\}$ ,其中 $w_{kj}$ 是文档 $d_j$ 中词 $t_k$ 的权重。

在向量空间模型中,文档表示有两个问题:为单词赋予权重和度量特征向量的相似度。常用的加权模式有基于文本实验观察结果的TF-IDF(词频—逆文档频率)[79]:

- 稀有词相关性不小于频繁词相关性(逆文档频率假设);
- 一篇文档中多处出现的词的相关性不小于只出现一次词的相关性;
- 长文档不一定好于短文档(归一化假设)。

换句话说,在一篇文档中频繁出现(TF=词频)但很少出现在语料库中其他的文档里(IDF=逆文档频率)的单词,与该文档主题的相关性可能很大。另外,结果权重向量的归一化防止了长文档有更好的检索机会的问题。TF-IDF函数很好地解释了这些假设:

$$\text{TF-IDF}(t_k, d_j) = \underbrace{\text{TF}(t_k, d_j)}_{\text{TF}} \cdot \underbrace{\log \frac{N}{n_k}}_{\text{IDF}} \quad (3.1)$$

$N$ 表示语料库中文档的个数, $n_k$ 表示含有词单词 $t_k$ 出现至少一次的文档集合的数量。

$$\text{TF}(t_k, d_j) = \frac{f_{k,j}}{\max_z f_{z,j}} \quad (3.2)$$

最大值是出现在文档  $d_j$  中所有单词  $t_z$  的词频  $f_{z,j}$  上计算的。为了使权重落在  $[0, 1]$  区间且文档能够用等长向量表示, 利用式(3.1)获取权重通常用余弦归一化方式来归一化:

$$w_{k,j} = \frac{\text{TF-IDF}(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|} \text{TF-IDF}(t_s, d_j)^2}} \quad (3.3)$$

其实现了归一化的假设。

如前所述, 一个相似度的度量需要先确定两个文档的接近程度。许多相似度的度量都衍生于对两个向量的接近程度的度量; 在这些度量中, 余弦相似度被广泛应用:

$$\text{sim}(d_i, d_j) = \frac{\sum_k w_{ki} \cdot w_{kj}}{\sqrt{\sum_k w_{ki}^2} \cdot \sqrt{\sum_k w_{kj}^2}} \quad (3.4)$$

在依赖于向量空间模型的基于内容的推荐系统中, 用户个人信息和物品都被表示成带权重的词向量。预测用户对一个特定物品的兴趣可以通过计算余弦相似度得到。

### 3.3.1.2 基于关键词系统概述

许多基于关键词的推荐系统发展时间相对较短, 但在很多领域都可以发现它们在应用程序中的应用, 如新闻、音乐、电子商务、电影等。每个领域面对不同的问题, 也就需要不同的解决方案。

在 Web 推荐系统的领域中, 关于内容的著名系统有 Letizia[49]、Personal Web-Watcher[62, 63]、Syskill & Webert[70, 68]、ifWeb[4]、Amalthea[66] 和 WebMate[23]。Letizia 是一个网页浏览器的扩展, 它跟踪用户浏览行为, 并依据与用户兴趣相关的关键词进行个性化建模。它依赖隐式反馈来推断用户的喜好。例如, 收藏一个网页就表示为用户对这个网页感兴趣的有力证据。同样的方式, 个人 WebWatcher 从用户访问的网页和离开已访问网页的链接来学习个体的兴趣。它将访问的网页作为用户兴趣的正例, 未访问的网页作为负例。Amalthea 使用指定的过滤工具来协助用户发现感兴趣的信息。用户可以通过提供与用户兴趣相关的网页(表示为权重向量)来确定过滤器。

Syskill & Webert 采用了相同的方法, 它用 128 个最有代表性的词(文档中有代表性的词可以用许多不同的方法确定)来表示文档。ifWeb 采用了更高级的表示技术, 它将信息表示成一定形式的带权重的语义网。它支持显式反馈, 并且不仅考虑兴趣, 而且还考虑了显式的非兴趣。另一个有趣的方面是, 它加入了一个时间衰减的机制, 如给表示用户的兴趣加上时间衰减。WebMate 采用了另一个不同的方法来表示用户兴趣, 它在不同领域通过学习由正向训练样例表示的关键词向量构成的用户信息来跟踪用户兴趣。一个  $n$  个关键词的向量可以正确地表示最多  $n$  个独立的用户兴趣。

在新闻过滤领域, 著名的推荐系统有 NewT[87]、PSUN[90]、INFOrmer[91]、NewsDude[12]、Daily Learner[13] 和 YourNews[2]。NewT(News Tailor) 允许用户提供关于文章、部分章节、作者或来源的显式和隐式反馈。许多过滤工具用不同类型的信息进行训练, 如政治类新闻过滤器、体育类新闻过滤器等。YourNews 是一个最近新出现的个性化新闻访问系统, 使用同样的方法为 8 个不同主题(国内、国际、经济等)分别维护有一个兴趣个人信息。用户对这些主题的兴趣信息用一个加权的原型词向量表示, 向量值从用户浏览的新闻的历史记录中抽取。用户过去浏览了  $N$  篇文章, 抽取前 100 个加权的词

来生成用户最终的原型向量。该系统维护了一个仅考虑最近浏览过的 20 篇新闻信息的短期特征，而长期特征考虑的是过去所有浏览过的新闻。系统可以利用这些个人信息来显示最近的和推荐的两类新闻。<sup>①</sup>

在新闻过滤系统中，学习短期和长期特征是两种非常典型的方式。NewsDude 在用户提供的感兴趣新闻的初始训练集基础上，用基于 TF-IDF(余弦相似度)的方法学习用户短期模型，用基于朴素贝叶斯分类器的模型学习用户的长期模型。新闻来源于雅虎新闻。同样的 Daily Learner 无线信息访问的学习工具，也采用一种方法来学习两种不同的用户模型。前者基于最近邻的文本分类器算法来维护用户的短期兴趣特征，后者利用长时间收集到的数据，基于朴素贝叶斯分类器，来学习用户的长期兴趣特征。

在对文章和个人信息使用更复杂表示的系统中，需要注意 PSUN 和 INFOrmer。PSUN 采用了一个可选择的文本表示方法。初始特征由系统中用户选择感兴趣的某些文章来提供。文本中重复出现的单词被记录在称为 n-gams 的网络中，这个网络中的单词可以相互吸引或者排斥，相互吸引程度主要取决于网络中两个词共同出现的次数。每个用户都有多个特征，它们是需要显式反馈，经过一个遗传算法来竞争产生的。INFOrmer 使用一个语义网络来同时表示用户特征和文本。一个扩散激活技术[25]用来比较文本和用户特征，并且为使系统行为适应变化的用户兴趣，采用一个相关的反馈机制。一个纯扩散激活模型是由被标记或加权的链接起来的节点组成的数据网络。给一组源节点标记激活权重，然后不断迭代，将源节点激活权重传播给与它相连的其他节点，直到终止条件满足使得网络中的搜索过程结束。

各种不同的基于内容的推荐系统在其他的应用领域也有应用。LIBRA[65]做的书籍推荐，利用从 Amazon 在线电子商店获取的关于产品描述的网页，实现了一个朴素贝叶斯文本分类方法。Re: Agent[16]：智能的电子邮件工具，它利用自动的特征抽取算法，可以学习如过滤、下载到掌上电脑、将邮件转为语音邮件等行为。Re: Agent 用户只需要将样例信息放置在与期望行为对应的文件夹中。Re: Agent 从这些文件夹中学习概念和决策策略。Citeseer[15]通过使用文字信息和分析论文中的共同引文，来协助用户搜索学术文献。INTIMATE[53]使用文本分类技术从 Internet Movie Database<sup>②</sup>获得的电影剧情简介去学习，进而推荐电影。为了获得推荐电影，用户被要求至少给一定数量的电影进行评价，评价分为 6 个档次：很差、差、低于平均、高于平均、好、优秀。同样，Movies2GO[67]从用户评分过的电影剧情简介中学习用户偏好。该系统的创新之处是加入了投票模式[93]，当多人在偏好上有冲突的时候，给一个可接受的折中，使他们的冲突偏好能更好地适应单个用户。

在音乐领域，一般使用的推荐技术是协同过滤(见 Last.fm<sup>③</sup>和 MyStrands<sup>④</sup>系统)。最著名的系统 Pandora<sup>⑤</sup>使用(人工的)基于内容的描述来推荐音乐。该系统的主要问题是可扩展性，因为音乐注释的过程全是靠人工完成的。相反，FOAFing the music[21, 22]可以推荐音乐、发现音乐和探索音乐的内容，它基于来自 Friend of a Friend(FOAF<sup>⑥</sup>)的用户信息、从与音乐相关的 RSS 种子中提取的上下文信息，以及自动地从音频文件中提取

① 这里的“recent and recommended”应该表示的是两种推荐的形式。——译者注

② <http://www.imdb.com>

③ <http://www.last.fm>

④ <http://www.mystrands.com>

⑤ <http://www.pandora.com>

⑥ <http://www.foaf-project.org>



的基于内容的描述。

为了完成采用简单的关键字向量空间表示的基于内容的推荐系统的研究，我们也会提到一些融合了协同和基于内容的方法的混合推荐系统，如 Fab[7]、WebWachter[45]、ProfBuilder[99]、PTV[89]、Content-boosted Collaborative Filtering[56]、CinemaScreen[78]和在文献[94]中提到的一个。

对过去 15 年里主流系统的发展进行分析学习，最重要的发现是，同时对物品和用户信息采用基于关键词的表示，并通过足够多的信息证明用户兴趣可用，可以准确预测用户行为。大多数基于内容的系统被认为是建立在包括用户兴趣正例和负例文本训练集上的文本分类器。因此推荐的准确率依赖于大量的训练样本，而训练样本的好坏又依赖于对用户兴趣的可靠的句法分析的结果。这个方法的问题就是“缺乏智能”。当要求更高级的特性时，基于关键词的方法就显得不足。如果用户有个如“法国印象派”的关键词，基于关键词的方法只能找出包含有词“法国”和“印象派”的文档。那么关于 Claude Monet 或 Renoir 展览的文档将不会出现在推荐集合中，即使他们可能对用户来说更相关一些。更高级的表示策略需要为基于内容的推荐系统添加“语义智能”，它可以超越由关键词提供的用户兴趣的语法证据。

下面将会探究在索引阶段通过本体和广博知识源注入知识的可能方式。

### 3.3.1.3 运用本体的语义分析

语义分析需要学习大量准确的信息，这些信息包括参考在外部知识基础上定义的概念。这个方法的主要的目是提供一个有文化和语言背景知识的推荐系统，且系统具有翻译自然语言文档和分析内容的能力。

本节将会介绍一些现在主流策略里所采用的语义推荐过程。这些策略大致主要考虑以下几个标准：

- 包含知识源的类型(如词典、本体等)；
- 物品的注释或表示所采用的技术；
- 用户个人信息中内容的类型；
- 物品-用户个人信息匹配策略。

SiteIF[52]是一个多语言新闻网站的个性化的工具。为了最好地表示知识，这是第一个采用基于感知的文档表示来对用户兴趣建模的系统。MultiWordNet 是一个在表示过程包括外部知识源的系统和一个集合了英语和意大利语的多语言词汇数据库。每篇新闻会使用词领域消歧(Word Domain Disambiguation)[51]来自动关联到一个 MultiWordNet 的同义词集合。用户信息是来自于用户读过的文档中发现的语义网络中某节点表示的同义词集合。在匹配阶段，系统将文档的同义词表示集合和当前用户模型作为输入，将使用语义网络数值技术(Semantic Network Value Technique)[92]产生的相关文档作为预测输出。

ITR(Item Recommender)是一个可以在多个领域应用的物品推荐系统(如电影、音乐、书籍)，它以文本的形式提供了对物品的描述(如情节摘要、概述、短摘要)[27, 83]。和 SiteIF 相似，ITR 在学习用户信息的过程中集合了多语言知识，但是与词领域消歧不同，词感知消歧(Word Sense Disambiguation)采取的是基于感知的文档表示。多语言知识不包含来自于 WordNet 词汇的本体。物品根据一个基于同义词集合的向量空间模型进行表示，称作同义词袋(Bags-of-Synsets, BOS)，它是经典的词袋模型(Bags-of-Words, BOW)的一个扩展[8, 84]。在 BOS 模型中，同义词集合向量不是像一个词向量对应一篇文档一样。用户信息采用一个朴素贝叶斯二进制文本分类器可以将文本分类成感兴趣和不感兴

趣。在训练阶段,根据条件概率估计值,可以得到更能象征用户偏好的那些同义词集合。文本—用户信息匹配操作在于通过使用用户信息的同义词集合的概率,计算物品属于“感兴趣”类的概率。

SEWeP(Semantic Enhancement for Web Personalization,用于Web个性化的语义增强)[31]是一个Web个性化系统,它利用日志和Web站点内容的语义来实现个性化。为得到统一一致的词汇表,采用了特定领域的类别分类对网页进行语义标注。类别是人工建立的,而标注的过程是自动完成的。SEWeP,像SiteIF和ITR一样,利用WordNet中的词典知识,来“翻译”文本的内容并支持标注/表示的过程。网页最初被表示成从其内容中抽取的关键词,然后这些关键词被映射到类别中的概念。给定一个关键词,基于WordNet中词的相似度度量,用来找出与这个关键词“最近”的类别的词。SEWeP不会给用户创建个性化信息,而是发现一种导航模式。SEWeP的推荐引擎利用与一个模式有“语义关联”的类别来扩展个性化网页推荐集合,这个类别可能就是用户感兴趣的主体分类。

Quickstep[58]是一个在线学术研究论文的推荐系统。该系统是根据网站DMOZ上的开源目录工程<sup>①</sup>(27个类别)的计算机科学分类来抽取学术论文主题的本体。运用k-NN分类器结合论文的语义标注与学术论文主题本体。用户的兴趣信息从之前浏览的学术论文所属分类的关联计算得到。因此,用户兴趣特征就是一组主题及主题的兴趣值。物品—用户特征匹配通过计算用户个人信息和被分为该类的论文的前三个感兴趣的主体来实现。除了网页推荐界面,Foxtrot[58]通过实现论文查询接口、个人信息可视化接口和邮件提醒来扩展Quickstep系统。特征被表示成一组用户可以理解的本体,使用户特征可视化成为可能。

Informed Recommender[1]使用消费者产品评价给出推荐建议。系统通过作为知识表示和共享的翻译本体,把用户观点转换成一个结构化形式。本体提供了一个可控的词表及要描述的词与词的关系:消费者的专业水平及对被评论的产品的使用经验。为了这个目的,本体主要包含了两部分:观点的质量和产品的质量,由此形成了两个形式化概念。一个文本挖掘过程自动地将评价中的句子映射到本体信息结构。该系统不会为用户建立特征,而是基于用户请求计算推荐集,如用户对产品特定特征的质量有所要求。Informed Recommender可以响应这个查询,并且可以根据用户关心的特征属性推荐最好的产品。值得注意的两个方面:一个是本体知识可以根据产品的标注对不同观点建模;另一个是用户的评价描述是以自由文本的形式。

News@hand[18]是一个采用基于本体表示物品特征和用户偏好来推荐新闻的系统。标注过程将新闻和所属领域的本体联系起来。他们采用IPTC<sup>②</sup>囊括了多个领域的共17个本体库,如教育、文化、政治、宗教、科学、体育等。但并不清楚它是用人工还是用像文本分类一样的自动化技术进行标注的。物品的表示是定义在本体概念空间的TF-IDF得分的向量。用户特征也用相同的空间进行表示,但分值度量的是用户对特定概念的兴趣强度。物品—用户特征匹配通过计算向量的余弦相似度得到。

之前提到过的推荐系统Interactive Digital Television[14],借鉴语义网的合理技术来比较用户偏好和物品(电视节目),比传统的语义指标更有弹性。在推荐过程中可用的电视节目使用能够正确描述其主要属性的元数据进行标注。电视领域的知识和用户特征都采用

① <http://www.dmoz.org/>

② <http://nets.ii.uam.es/neptuno/ipct/>

OWL 本体来表示。本体—用户特征给出了用户偏好的表示, 可以找出偏好的原因, 也可以发现跟用户兴趣有关的额外知识。推荐阶段采用存储在用户特征中的知识来发现隐藏在用户偏好和产品之间的语义关联。知识推理的过程和扩散激活技术被应用在给用户建议产品。这项工作值得注意的方面是, 基于扁平列表的方式不是良好结构化的数据, 本体—用户特征改进了这种方式, 帮助发现新的知识。

JUMP 系统[10, 9]可以智能地将基于情景的和个性化的信息发送给日常工作环境中做着非常规工作的知识工程师。JUMP 用户的信息需求被表示成复杂的查询, 查询可以是一个任务支撑需求, 而不是一个用户信息。比如, 一个复杂的查询例子“我需要准备一个 VIKEF 工程的技术报告”。同时对文档和用户需求信息进行语义分析, 是基于其中概念是通过 WordNet 同义词集合进行手工标注的领域本体。而从文档到领域或词典概念的映射是通过词感知消歧和命名实体识别自动完成的, 其中在领域本体中使用了词典标注。而用户需求的概念和文档的概念匹配是根据他们在领域本体中的联系。在前面的查询例子中, 概念“技术报告”和“工程”的实体及实体之间的关系都是从本体中提取的。

WordNet 采用了语义消歧技术, 经常用在语义翻译上, 而由于其广泛使用, 语言知识的领先地位越来越突出。另一方面, 上述研究表明, 仅有 WordNet 所带来的巨大潜力还不足以理解用户兴趣以及应用领域中兴趣的上下文。特定领域的知识是必需的。本体是应用领域形成的基础, 用于领域中的物品的语义描述、概念(如类别及其实例)的表示、关系(如层次关系、属性)的定义中。总之, 相比于传统的基于内容的方法, 不管是囊括了语言还是特定领域知识, 抑或是基于内容的过滤方法的研究, 都提供了更好、更准确的推荐结果。这也鼓励了研究人员利用像主题词表或本体一样的外部知识形式化用户兴趣并将其置于上下文中, 来设计研究用户兴趣的新颖的过滤方式。

#### 3.3.1.4 运用百科全书式的知识源进行语义分析

相比于只有很多词的集合, 常识和特定领域知识产生更有信息量的特征, 对于改进自然语言处理技术可能会有帮助。在学习用户特征的过程中, 相对于经典的使用内因知识的技术, 注入外因知识(外部提供的)得到的效果会更好。近些年来, 世界上很多的知识信息源已经变得容易获取。举例来说, 开放目录工程(Open Directory Project, ODP)、Yahoo! Web Directory、Wikipedia 都是通用的知识库。

接下来简单地回顾下利用通用知识产生新的高级特征的新方法, 即使其中有些方法在学习用户特征的情景中并没有被使用。

显式语义分析(Explicit Semantic Analysis, ESA)[34, 35]技术利用 Wikipedia 中的概念将自然语言文本表示成细粒度的语义概念。Wikipedia 的文章定义了概念, 如 Italy、计算机科学、推荐系统。这类方法是受增加对大量客观世界知识进行文本表示的需求而产生的。将 Wikipedia 作为知识源有几个好处, 如社区成员的不断完善、有多种语言可使用, 及其高度准确性[37]。经验显示, ESA 对计算词语文本的相关性和在不同数据集上的文本分类都有本质的改进。这也表明 ESA 确实提高了传统基于 BOW 的检索模型[30]。

另一项有意思的方法是 Wikify! 系统提出的给文本添加语义[59, 26], 也就是可以识别文本中的重要概念(关键词抽取), 并将这些概念和相应的 Wikipedia 网页相连(词义消歧)。Wikify! 系统给出的这些标注可以自动地增加与文本内容语义相关的信息。为了比较系统给出的标注和 Wikipedia 贡献者提供的人工标注的质量, 研究人员设计了一个类似图灵测试的试验, 要求参与者区分出人工和自动标注。结果显示, 计算机和人工给出的标注很难区分, 即说明 Wikify! 系统的标注质量很高。

据我们所知,还没有推荐系统(基于内容)可以利用前面提到的高级语义文本表示的方法学习到用户现实世界的真实的特征。在几个任务中利用如语义关联、文本分类、检索等先进的文本表示方法获取到的正向结果显示在推荐任务中也能得到类似的正向结果。这看起来是一个有前景但还未被探讨的研究领域。

在文献[47]中,为了提高 Netfilx Prize 比赛的推荐准确度, Wikipedia 常被用来估计电影之间的相似度。更具体地, Wikipedia 文本的内容和超链接被用来衡量电影之间的相似度。基于包含每两部电影之间相似度的相似度矩阵,用户预测的估计值使用 K 近邻和伪 SVD 算法进行计算。这些方法都是将 Wikipedia 的相似度估计与训练集的评价合并,在测试集上进行预测估计。遗憾的是,这些技术对于整体的精确度的提高没有非常重要的改进。

文献[88]介绍了一个很复杂但还没有完成的过滤 RSS 种子和电子邮件的方法。更具体地,作者介绍了利用 Wikipedia 从用户文档集合中自动提取产生用户特征的方法。这个方法主要由 4 个步骤构成,即 Wikipedia 索引、特征产生、问题导向索引数据库建立和信息过滤。特征产生阶段利用隐式表示用户感兴趣主题的历史文档集合。从这些文档中抽取特征词,并使用 ESA 算法得到它们和 Wikipedia 相似的文章。这时系统从文章中抽取出 Wikipedia 的分类列表,聚合这些类别,从而得到对应于用户个人信息中一个主题的子类别。用户也可以查看自己的用户特征,并且可以增加和修改这些类别,进而改善主题特征。对于用户特征的每个主题,建立一个问题导向的 Wikipedia 语料库并索引,来表示过滤信息库。

文献[85]介绍了一个在内容分析阶段利用 Wikipedia 的不同方法。更确切地说,该方法是一个向基于内容推荐系统灌输知识的过程,希望通过增加文化背景知识,相比于经典的基于词的方法,提高内容分析的准确性。百科全书式的知识有利于识别特定的依赖领域的概念或命名实体,尤其是在那些领域本体应用不可行的上下文中。Wikipedia 的实体在词空间模型的基础上一直使用语义向量进行建模[77],该模型基于词空间、向量空间的点表示一组语义概念,如词和文档。词之间的关系通过使用扩散激活算法产生新特征,这些新特征可以用在推荐过程中的多个方面。

### 3.3.2 学习用户特征的方法

通常用于推导基于内容的用户个人信息的机器学习技术也会非常适合用于文本分类[82]。在用来文本分类的机器学习方法中,推导过程通过从一个训练文档集合(文档已经被标记为他们所属的分类)中学习类别的特征,自动建立一个文本分类器。

学习用户特征的问题可以转换为一个二元文本分类任务:每一个文档都根据用户的偏好被分类成感兴趣或不感兴趣。因此,类别集合为  $C = \{c_+, c_-\}$ , 其中,  $c_+$  是正类(用户喜欢的),  $c_-$  是负类(用户不喜欢的)。

下面将回顾在基于内容的推荐系统中使用最多的学习算法。它们能够学习一个能对每个用户兴趣建模的函数。这些方法通常要求用户赋给文档一个相关的分数来标记文档,自动推断出用户的个人信息,并用于过滤过程根据用户的偏好对文档排序。

#### 3.3.2.1 概率方法和朴素贝叶斯

朴素贝叶斯是一个归纳式学习的概率方法,属于一般的贝叶斯分类器。这类方法基于之前的观察数据产生一个概率模型。该模型估计文档  $d$  属于类别  $c$  的后验概率  $P(c|d)$ 。

这个估计是基于先验概率  $P(c)$ ，即观测到一个文档属于类别  $c$  的概率， $P(d|c)$  即在给定类别  $c$  的情况下观测到文档  $d$  的概率，和  $P(d)$  即观测到文档  $d$  的概率。使用这些概率，应用贝叶斯定理来计算  $P(c|d)$ ：

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)} \quad (3.5)$$

为了对文档  $d$  分类，选择概率最高的作为类别：

$$c = \operatorname{argmax}_{c_j} \frac{P(c_j)P(d|c_j)}{P(d)}$$

$P(d)$  与所有类别  $c_j$  相等时，一般将其去掉。当我们不知道  $P(d|c)$  和  $P(c)$  的值时，我们利用观测到的训练数据对它们进行估计。然而，这样估计  $P(d|c)$  是有问题的，因为相同的文档不太可能再次出现：观测数据普遍不足以产生好的概率。朴素贝叶斯分类器通过独立假设简化了该模型，从而克服了这个问题，该独立假设为：在观测文档  $d$  中，在给定类别下，所有的词或记号之间是条件独立的。文档中，这些词的个体概率是分别估计的，而不是将整个文档作为一个整体。条件独立的假设明显地违反了现实世界的的数据，尽管如此，朴素贝叶斯分类器在实际文本分类经验上确实做得不错[48, 11]。

有两个被普遍使用的朴素贝叶斯分类模型：**多元伯努利事件模型**和**多项式事件模型**[54]。两个模型都将文档看作一个在语料库词汇表  $V$  上的向量值，向量中的每个实体表示它在这个文档中是否出现，因此模型都损失了关于词顺序的信息。多元伯努利事件模型将每个词编码为一个二元属性，即一个词出现或没出现，而多项式时间模型计算一个词在一个文档中出现的次数。经验结果显示，多项式朴素贝叶斯公式的表现胜过多元伯努利模型。尤其在巨大的词汇表下，这个效果比较明显[54]。多项式事件模型计算  $P(c_j|d_j)$  方式如下：

$$P(c_j|d_i) = P(c_j) \prod_{w \in V_{d_i}} P(t_k|c_j) N_{(d_i, t_k)} \quad (3.6)$$

$N_{(d_i, t_k)}$  表示词或记号  $t_k$  在文档  $d_i$  出现的次数。这里要注意，仅是文档  $d_i$  中包含的词汇表子集  $V_{d_i}$  的概率相乘，而不是整个语料库中词汇表  $V$  中所有的词的概率相乘。

实现朴素贝叶斯的一个关键步骤是估计词的概率  $P(t_k|c_j)$ 。为了使概率估计对于很少出现的词更有健壮性，需要采用一个简单的事件计数的平滑方法修正这个概率。一个很重要的平滑作用就是它避免了在训练数据的某一类中，一个没有出现过的词概率为 0 的情况。一个相当简单的平滑方法是基于常用拉普拉斯估计(如一个类中，所有的词计数都给它加 1)。Witten-Bell[100]是另一个更有趣的方法。尽管朴素贝叶斯的表现不如其他的统计学习的方法，如基于近邻的分类器或者支持向量机，但它在那些对计算概率要求不那么高的分类任务上表现得非常好。朴素贝叶斯方法的另一个好处是它的高效和相比于其他学习方法更容易实施。

尽管基于多项式的模型分类器在词汇表数量巨大的情况下，优于基于多元的模型，他们的表现在以下情况下也不令人满意：1) 训练集合中的文档不一样长，因此导致参数估计粗糙；2) 处理稀少的类(很少的训练样本可用)。这些情况都频繁出现在用户特征建模的任务中，其中训练文档的长度无法给出假设，获取合适大小的负例样本(如类别  $c_-$  的样本)是个问题。事实上，用户不会马上察觉到向系统提供负反馈时效果会立竿见影[81]，训练集类别  $c_+$  样本(用户喜欢的)可能经常都比类别  $c_-$  (用户不喜欢的)更大。在文献[46]中，作者提出了一个多元泊松模型的朴素贝叶斯文本分类器，它采取比前面提到的条件概率更合理的参数估计。我们已经将该方法用到[36]中提到的用户特征任务中。



朴素贝叶斯分类器用在许多基于内容的推荐系统中, 如 Skill&Webert[70, 68]、NewsDude[12]、Daily Learner[13]、LIBRA[65]和 ITR[27, 83]。

### 3.3.2.2 相关反馈和 Rocchio 算法

相关反馈是一个信息检索中应用的技术, 它帮助用户逐步完善基于之前搜索结果的查询。它是由用户根据所需信息检索到的相关文档, 给出的进一步用户反馈组成的。

相关反馈及其在文本分类中的融合和著名的 Rocchio 公式[75], 都普遍被基于内容的推荐系统所采用。基本的原则是允许用户给推荐系统根据用户的信息需求推荐的文本打分。这种形式的反馈随后被用来逐步改善用户特征, 或者为训练将用户个人信息作为分类器的学习算法所使用。

一些线性分类器由类别的显式描述(或原始文档)组成[82]。Rocchio 算法是一个可用于推导出这种线性的且带有描述风格的分类器。该算法将文档表示成向量, 因此有相似内容的文档有相似的向量。向量的每个成分对应着文档中的一项, 比较典型的是一个词。而每个成分的权重使用 TF-IDF 计算得到。在类别集合  $C$  中, 对每个类, 通过合并文档向量(正例和负例样本)得到每个类的原型向量来完成学习过程。对新文档  $d$  分类, 计算文档  $d$  表示的向量与每个类的原型向量的相似度, 然后把  $d$  分配给与原型相似度最高的类。

更正式地, Rocchio 算法计算一个分类  $c_i$  的向量  $\vec{c}_i = \langle \omega_{1i}, \dots, \omega_{|T|i} \rangle$  ( $T$  是词汇表, 训练集里不用词的集合), 公式如下:

$$\omega_{ki} = \beta \cdot \sum_{\{d_j \in POS_i\}} \frac{\omega_{kj}}{|POS_i|} - \gamma \cdot \sum_{\{d_j \in NEG_i\}} \frac{\omega_{kj}}{|NEG_i|} \quad (3.7)$$

其中  $\omega_{kj}$  是文档  $d_j$  中词  $t_k$  的 TF-IDF 权重,  $POS_i$  和  $NEG_i$  是指定类别  $c_i$  中的正例和负例样本集合。 $\beta$  和  $\gamma$  是控制参数, 可以用来设置所有正例和负例样本的相关程度。计算文档向量  $\vec{d}_j$  与每个类别的原型向量  $\vec{c}_i$  的相似度, 和  $c_i$  相似度最高的类别为  $\tilde{c}$ , 则将  $\tilde{c}$  作为文档  $\vec{d}_j$  的类别。基于 Rocchio 的分类算法没有任何理论基础, 也不保证有效或收敛[69]。

相关反馈已经在许多基于内容的推荐系统中使用, 如 YourNews[2]、Fab[7] 和 NewT[87]。

### 3.3.2.3 其他方法

用在基于内容的推荐系统中的还有其他的学习算法, 接下来简要地介绍一些非常重要的算法。完整的概述在[64, 69, 82]中有介绍。

在决策树里, 所有内部节点用词来标记, 从它们出发的分支被标记为对测试文档中词的权重的测算, 叶子节点用类别标记。决策树的学习是通过递归地分割训练数据(即文本文档)为子数据集, 直到这些子数据集仅包含一个类别为止。数据分割是把文档所含将要标记的词作为内部节点, 在其上做权重计算的测算得到的。选择哪个词作为分割的操作, 普遍采用的是信息增益或信息熵准则[101]。Syskill&Webert[70, 68]推荐系统使用的是决策树。

决策规则分类器和决策树类似, 他们都采用上面介绍的相似的方法递归地分割数据。规则学习器的一个优点是, 相比于决策树学习器, 它可以生成一个更简洁的分类器。规则学习的方法常试图从所有可能的规则(如可以正确将训练样本全部正确分类的规则)中根据一些最小化准则挑选“最好”的规则。

最近邻算法, 也称为懒惰学习器, 将训练数据简单地存储在内存里, 对于一个新的样本, 通过一个相似函数比较它与存储的所有样本的相似度, 从而对它进行分类。确定“最

近邻”或“k-最近邻”物品，未分类的类标签来源于最近邻的类标签。这时就需要相似度函数，例如，当样本用 VSM 表示的时候，采用余弦相似度。最近邻算法非常有效，虽然其最大的缺点是在分类时效率低下，因为缺少一个真正的训练阶段，所以拖延了分类时间。Daily Learner[13]和 Qucikstep[58]使用最近邻算法建立一个用户短期兴趣模型，并将论文的语义标注和类别名中的本体分别关联起来。

### 3.4 趋势和未来研究

#### 3.4.1 推荐过程中用户产生内容的作用

Web 2.0 是一个描述万维网技术趋势的术语，万维网的目标是促进用户之间的信息共享和协作。按照 Tim O'Reilly<sup>①</sup>说法，术语 Web2.0 的意思是以用户为中心，设计用户产生内容的软件，因为其内容是由成千上万用户所贡献的，如 Flickr, Wikipedia, Del.icio.us, YouTube 等网站。这也是 Web2.0 被称为“参与性 Web”的原因。同时，O'Reilly<sup>②</sup>也定义 Web2.0“为了让更多用户能使用的、更好的系统设计方法”。

在科研领域受到许多关注的一种用户产生内容(User Generated Content, UGC)的方法是大众分类法，这是一种由用户产生的分类学，用户大量地选择被称为标签的关键字来协助标注和分类感兴趣的资源。

尽管在推荐系统领域里已经有相当数量的重要研究，但是相比推荐标签问题，(如辅助用户进行标注)[98, 95]，人们对将标签整合到标准推荐系统算法，特别是基于内容的算法，所面临的特定问题的探索还少很多。

大众分类法在推荐系统领域提供了新的机会和挑战(参考第 19 章)。我们应该研究这些标签是否能够代表用户兴趣的有价值信息，是否应该被包含在用户信息中。当前已经发现了设计标签系统的几个难点，如多义词、标签同义词，或者是不同领域的标注者同时标注，会使得标签在描述资源对象的时候处于多个抽象层次，或者造成混乱的标签扩散现象[40]。

##### 3.4.1.1 推荐系统的社会化标签

现有研究中已经在基于内容的推荐系统中引入了几种用户打标签的方法。

在文献[28]中，用户个人信息以标签向量的形式展现，每一个元素表示一种标签被用户赋值给文档的次数值。文献[57]中提出了一个更加复杂的方法，考虑了标签的共现。用户个人信息和信息源的匹配是通过使用简单字符串匹配来实现的。正如作者本身所预见到的，采用 WORDNET 可以改进匹配效果。

在由 Szomszor 等[96]给出的文章中，作者描述了一种电影推荐系统，纯粹建立在采用协同标注方法赋给电影的关键字上。对于活跃用户，推荐系统的算法是基于电影相似性来计算的，即计算其他电影和用户评价过的电影之间的标签相似性。正如作者所述，推荐算法能通过综合基于标签属性和更多传统基于内容的推荐策略中的技术来提高推荐效果。

文献[33]提出几种不同的策略来建立基于标签的用户个人信息，并利用它们来构造音乐推荐系统。基于标签的用户个人信息是由标签集合来定义的，这些标签被用户用来标注唱片，同时标签也带有相应的分数用以代表用户在这些标签上的兴趣程度，如根据标签的

① <http://radar.oreilly.com/archives/2006/12/web-20-compact.html>, Accessed on March 18, 2009

② <http://www.npr.org/templates/story/story.php?storyId=98499899>, Accessed on March 18, 2009

使用频率和听音乐的频率来标注。

以上描述的方法中只有一组单一的热门标签来描述用户的兴趣,文献[102]提到这可能并不是最合适的描述用户的方法,因为它不能反映用户的多种兴趣。因此,作者提出一种网络分析技术(基于聚类方法),基于用户个性化标签来识别他们不同的兴趣。

关于标签解析的方法,Cantador等在文献[19]提出了一个模型来选择有意义的标签,利用WORDNET、维基以及谷歌等渠道获取一组初始未加工的标签,并从中选择合适的标签。如果标签在WORDNET中被直接匹配,那么直接使用;否则,可能的拼写错误以及复合名词通过使用谷歌的“你是×××意思吗”机制来发现匹配对象(例如,标签san-francisco或者san farncisco被纠正成san francisco)。最后,标签被关联到他们相应的维基条目上。

在Gemmis等的文章[36]中提出了一种更复杂的算法,算法实现了一种混合策略,从(静态)内容以及被用户评分的相关物品中学习用户个人信息。作者将用户个人信息,以及他自己的个性化标签包含进来,同时还采纳被其他用户评价过的物品的标签(社会标签)。当对大众化分类有贡献的用户具有不同的专业领域知识时,这方面显得特别重要。在用户个性化属性中引入社会标签的内容使得纯基于内容推荐的理论被扩展到混合协同内容理论[17]。此外,从标签中识别用户兴趣的解决方法也被提出来。由于用户自由选择标签以及他们实际行为不是很清楚的事实,作者建议通过基于WORDNET的词义消歧算法来解释标签。一个类似的混合方法也在文献[38]中被提出来,其方法结合了基于标签标注行为所表达的兴趣内容以及基于内容的用户个人信息。

文献[20]也提出了一些关于如何通过WORDNET方法来分析标签以便于获得用户意图的想法,但是实验结果并不支持文中提出的思路。文献[104]提出通过WORDNET的方法来语义解释标签的另一种方法。作者设计了一个让邻居进行选择的算法,证明了在协同过滤算法中标签的有效性。邻居选择是指在WORDNET中计算不同用户之间的标签语义距离。

我们相信,依靠不同知识源的方法(如WORDNET或维基)来识别标签含义的挑战性研究是有用的。此外,通过考虑个性化,社会化以及专家的标签的不同特性,可以设计出新的策略,在学习基于内容的用户个人信息过程中整合标签。事实上,个性化标签大部分是主观的,而且前后不一致,相反,专家的标签是客观和一致的。社会化标签则趋向某种形式的连贯性[5]。

另一种有意义的研究方向是以分析标签来获得有力的反馈来推断用户个人信息。表达用户观点和情绪的标签能表达用户对物品的满意度,如厌烦、感兴趣、好、坏等。情感计算研究领域的技术的确有必要。

### 3.4.2 超越特化:惊喜度

正如在3.2.2节中介绍的,基于内容的推荐系统被诟病过于特化,因为只推荐被用户评价过的相似物品。解决这一问题的可能方案是引入一些随机性。例如,信息过滤领域[87]曾提出过采用遗传算法。另外,特化问题不仅是避免推荐和用户以前用过的一样的物品。在一定情况下,如果物品和用户以前用过的物品太相似也不应该被推荐,例如,描述同一个事件的不同文章。因此,如果物品和用户以前用过的物品特别相似,一些基于内容的推荐系统会过滤掉这些物品,如Daily-Learner[13]。Zhang等[103]提出的冗余度量被用来衡量相关的文档是否包含新颖的信息。总之,推荐的多样性是推荐系统中通常的理想

特性。

在推荐系统中惊喜度可以被看作接收到未预料和意外的物品推荐的体验，因此是一种多样性推荐的方式。根据 Gup 的理论[41]，当用户依赖探索和运气来寻找他们不知道并且期望的物品时，基于内容的系统由于特化问题，在产生意外发现方面存在软肋。

明确区分新颖性和惊喜度之间的区别是有用处的。根据 Herlocker[42]的解释，新颖性是指系统建议给用户一个他已经能独立发现的未知物品。意外发现是指推荐系统帮助用户找到使用其他方法无法发现的一个令人惊讶的有趣物品(或者是很难真正被发现)。为了提供清晰的区分新颖性和惊喜度之间的区别的例子，请设想推荐系统只推荐用户喜欢的导演执导的电影。如果系统推荐一部用户未意识到的电影，其可能是新颖的，但是可能不是惊喜度。另一方面，推荐一部由一个新导演执导的电影可能更会提供一个意外惊喜。惊喜度的推荐按照定义也是新颖的。

惊喜度问题的挑战在于要去设计惊喜，也就是说要找到一种方式，能够在推荐过程中以一种可操作的方式推导出惊喜之处。从该角度而言，问题还没有得到深入的研究，并且理论和实验研究也都相当少。

如 Toms 解释的那样[97]，搜索信息有三种类型：

- 寻找已定义好对象的信息；
- 寻找不能完全描述对象的信息，但是第一眼能被识别的；
- 以一种碰巧、偶然或意外的方式获取信息。

不难看出，意外发现对于前两种获取方式没有用处，但是对于第三种方法特别重要。我们讨论关注于在基于内容推荐中发现意外惊喜策略的实现，对应到实际世界中的恰当比喻是，一个人去购物或者参观博物馆，当他漫无目的行走时，可能发现他从来没有期望发现过的全新东西，这个对他而言肯定有趣。在关于“可操作地推导出意外惊喜”方面已经提出不同的方法中，Toms 建议了四种策略，从最简单的到比较复杂的列举如下[97]：

- 碰运气或者撞大运，通过随机信息节点产生来实现；
- Pasteur 法则(机会垂青有准备的人)，通过用户个人信息来实现；
- 不寻常情况和例外情况，通过弱相似度量来部分实现；
- 类比推理，这些实现目前未知。

文献[44]，描述了一个实现“不寻常情况现象和异常例外”方法的提议，为了在传统的推荐中提供意外发现的推荐，因此在系统中提供用户新的指向物品的入口。最基本的假设是用户知道物品的概率越低，特定物品导致意外惊喜的概率越高。用户知道那些在语义上与系统确信用户所知物品相近的概率，要高于那些语义较远的物品的概率。换句话说，更加类似于通过提供与用户个人信息不相似的东西来获得意外发现。根据该原则，在文献[44]中提出的最基本想法是依据用户个人信息和物品描述之间的相似度来搜寻意外发现物品。系统依据朴素贝叶斯分类器来实现，它能够把物品分成感兴趣的( $C_+$ 类)和不感兴趣的( $C_-$ 类)，计算取决于由分类器计算好的先验概率。为了整合 Tom 的“低相似度”到推荐系统中，物品和用户个人信息的匹配会生成根据  $C_+$  类别中的先验概率产生的物品排列列表。列表的顶端是和用户个人信息最相似的物品，例如，被定为  $C_+$  类的最高评分物品。另一方面，被分类为  $C_-$  类的先验概率较高的物品在列表的底端。系统中不确定的物品是在两个分类分数之间趋于零的那些物品。因此假设这些物品不为用户所知是有意义的，因为系统不能够清晰地划分它们是否相关。那些在  $C_+$  和  $C_-$  之间区别很小的物品很难被分类，所以对于用户来说属于意外的发现。

在评估惊喜度问题的时候,很难获取惊喜的情感反应水平,因此有效的惊喜度衡量方法应该脱离传统的精确度度量方法以及相关的实验方法。文献[55]中列出了以用户为中心的评估推荐系统的新的指导性方向,如推荐系统的惊喜度。这些度量组成了有趣而且重要的研究课题(参考第8章)。

总之,在扩充基于内容的推荐系统的时候,采用具有惊喜度的策略,如提供令用户吃惊的推荐结果,可以有效降低系统的特化问题。

### 3.5 总结

本章总结了基于内容的推荐系统领域的相关研究,并且概述了系统所需要的各个重要特性与方法。尽管在不同的领域有许多的推荐系统,但是都用共同的方法来表示被推荐的物品和用户个人信息。本章首先讨论的主要论点涉及物品的表述,从最简单的结构化数据描述,到信息检索研究领域中比较复杂的非结构化数据。我们分析了过去15年推荐系统的主要内容,并强调一种更加复杂“语义分析”是必要的,这个方法的复杂性远超通过关键字来描述用户兴趣。概述了在推荐过程中采用的一些主要的语义策略,并提供了语言学知识主导性作用的证据,在不同的应用领域中基于不同语境的用户兴趣分析需要更加深入且专业的知识。最新的研究中也强调了运用世界范围的知识的重要性,如Wikipedia,尽管它们在学习用户个人信息的文本中还没有被使用。在总结中也提出了多种学习算法。

本章的最后一部分致力于讨论下一代基于内容推荐系统的主要趋势以及研究。特别是,本章提出了有关Web 2.0发展和变革的一些内容,将改变个性化的玩法,因为用户的角色从被动消费信息进化到主动贡献信息。最近的一些研究工作以及将来的研究方向中包含了一些有争论的策略,例如,把整合用户自定义词汇作为一种扩展词典的方式,如大众分类法。

最后,文中提出了内容推荐系统中非常特别的一个方面。由于这种系统本身的性质,他们只能推荐与用户个人信息匹配度很高的物品,因此用户被限制在那些与用户曾经评价过的物品相似的物品集合中。这种被称为特化的缺点阻碍了系统在真实世界场景中推广使用。文中分析了一些为用户提供惊讶和非预期(意外发现)结果的可能方法。

最后,我们想强调高级物品表示方法中语言处理研究的重要性。以下引用一个例子,我们从美国专利局和版权署中获取谷歌公司的专利消息是非常有价值的。例如,这些专利之一,名字叫开放用户信息描述方法,像如下方法描述用户信息:“我真的喜欢远足,特别是能在外野营几天的长途远足。室内活动我一点都提不起兴趣,我真的厌烦类似于园艺的室内活动。”使用智能的语言处理算法来发现用户的情绪(对短途旅行和园艺是喜欢或者不喜欢)和其他语言线索,系统可能会提供户外活动相关的广告给该用户,但是尽量避免提供其他业余爱好方向的广告[3]。

我们希望本章所提出的论点能够对科研领域的下一代基于内容推荐技术起到促进作用。

### 参考文献

1. Aciar, S., Zhang, D., Simoff, S., Debenham, J.: Informed Recommender: Basing Recommendations on Consumer Product Reviews. *IEEE Intelligent Systems* 22(3), 39–47 (2007)
2. Ahn, J., Brusilovsky, P., Grady, J., He, D., Syn, S.Y.: Open User Profiles for Adaptive News Systems: Help or Harm? In: C.L. Williamson, M.E. Zurko, P.F. Patel-Schneider, P.J. Shenoy (eds.) *Proceedings of the 16th International Conference on World Wide Web*, pp. 11–20. ACM (2007)
3. Anderson, M.: Google Searches for Ad Dollars in Social Networks. *IEEE Spectrum* 45(12), 16 (2008)



4. Asnicar, F., Tasso, C.: ifWeb: a Prototype of User Model-based Intelligent Agent for Documentation Filtering and Navigation in the World Wide Web. In: C. Tasso, A. Jameson, C.L. Paris (eds.) *Proceedings of the First International Workshop on Adaptive Systems and User Modeling on the World Wide Web, Sixth International Conference on User Modeling*, pp. 3–12. Chia Laguna, Sardinia, Italy (1997)
5. Aurnhammer, M., Hanappe, P., Steels, L.: Integrating Collaborative Tagging and Emergent Semantics for Image Retrieval. In: *Proceedings of the WWW 2006 Collaborative Web Tagging Workshop* (2006)
6. Baeza-Yates, R., Ribeiro-Neto, B.: *Modern Information Retrieval*. Addison-Wesley (1999)
7. Balabanovic, M., Shoham, Y.: Fab: Content-based, Collaborative Recommendation. *Communications of the ACM* **40**(3), 66–72 (1997)
8. Basile, P., Degemmis, M., Gentile, A., Lops, P., Semeraro, G.: UNIBA: JIGSAW algorithm for Word Sense Disambiguation. In: *Proceedings of the 4th ACL 2007 International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic, pp. 398–401. Association for Computational Linguistics (2007)
9. Basile, P., de Gemmis, M., Gentile, A., Iaquinta, L., Lops, P., Semeraro, G.: An Electronic Performance Support System Based on a Hybrid Content-Collaborative Recommender System. *Neural Network World: International Journal on Neural and Mass-Parallel Computing and Information Systems* **17**(6), 529–541 (2007)
10. Basile, P., de Gemmis, M., Gentile, A., Iaquinta, L., Lops, P.: The JUMP project: Domain Ontologies and Linguistic Knowledge @ Work. In: *Proceedings of the 4th Italian Semantic Web Applications and Perspectives - SWAP 2007, CEUR Workshop Proceedings*. CEUR-WS.org (2007)
11. Billsus, D., Pazzani, M.: Learning Probabilistic User Models. In: *Proceedings of the Workshop on Machine Learning for User Modeling*. Chia Laguna, IT (1997). URL [citeseer.nj.nec.com/billsus96learning.html](http://citeseer.nj.nec.com/billsus96learning.html)
12. Billsus, D., Pazzani, M.J.: A Hybrid User Model for News Story Classification. In: *Proceedings of the Seventh International Conference on User Modeling*. Banff, Canada (1999)
13. Billsus, D., Pazzani, M.J.: User Modeling for Adaptive News Access. *User Modeling and User-Adapted Interaction* **10**(2-3), 147–180 (2000)
14. Blanco-Fernandez, Y., Pazos-Arias J. J., G.S.A., Ramos-Cabrer, M., Lopez-Nores, M.: Providing Entertainment by Content-based Filtering and Semantic Reasoning in Intelligent Recommender Systems. *IEEE Transactions on Consumer Electronics* **54**(2), 727–735 (2008)
15. Bollacker, K.D., Giles, C.L.: CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications. In: K. Sycara, M. Wooldridge (eds.) *Proceedings of the Second International Conference on Autonomous Agents*, pp. 116–123. ACM Press (1998)
16. Boone, G.: Concept Features in Re:Agent, an Intelligent Email Agent. In: K. Sycara, M. Wooldridge (eds.) *Proceedings of the Second International Conference on Autonomous Agents*, pp. 141–148. ACM Press (1998)
17. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* **12**(4), 331–370 (2002)
18. Cantador, I., Bellogín, A., Castells, P.: News@hand: A Semantic Web Approach to Recommending News. In: W. Nejdl, J. Kay, P. Pu, E. Herder (eds.) *Adaptive Hypermedia and Adaptive Web-Based Systems, Lecture Notes in Computer Science*, vol. 5149, pp. 279–283. Springer (2008)
19. Cantador, I., Szomszor, M., Alani, H., Fernandez, M., Castells, P.: Ontological User Profiles with Tagging History for Multi-Domain Recommendations. In: *Proceedings of the Collective Semantics: Collective Intelligence and the Semantic Web, CISWeb2008*, Tenerife, Spain (2008)
20. Carmagnola, F., Cena, F., Cortassa, O., Gena, C., Torre, I.: Towards a Tag-Based User Model: How Can User Model Benefit from Tags? In: *User Modeling 2007, Lecture Notes in Computer Science*, vol. 4511, pp. 445–449. Springer (2007)
21. Celma, O., Ramírez, M., Herrera, P.: Foafing the Music: A Music Recommendation System based on RSS Feeds and User Preferences. In: *6th International Conference on Music Information Retrieval (ISMIR)*, pp. 464–467. London, UK (2005)
22. Celma, O., Serra, X.: FOAFing the Music: Bridging the Semantic Gap in Music Recommendation. *Web Semantics* **6**(4), 250–256 (2008)
23. Chen, L., Sycara, K.: WebMate: A Personal Agent for Browsing and Searching. In: K.P. Sycara, M. Wooldridge (eds.) *Proceedings of the 2nd International Conference on Autonomous Agents*, pp. 9–13. ACM Press, New York (1998)

24. Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., Sartin, M.: Combining Content-Based and Collaborative Filters in an Online Newspaper. In: *Proceedings of ACM SIGIR Workshop on Recommender Systems* (1999). URL [citeseer.ist.psu.edu/claypool99combining.html](http://citeseer.ist.psu.edu/claypool99combining.html)
25. Collins, A.M., Loftus, E.F.: A Spreading Activation Theory of Semantic Processing. *Psychological Review* **82**(6), 407–428 (1975)
26. Csomai, A., Mihalcea, R.: Linking Documents to Encyclopedic Knowledge. *IEEE Intelligent Systems* **23**(5), 34–41 (2008)
27. Degemmis, M., Lops, P., Semeraro, G.: A Content-collaborative Recommender that Exploits WordNet-based User Profiles for Neighborhood Formation. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research (UMUI)* **17**(3), 217–255 (2007). Springer Science + Business Media B.V.
28. Diederich, J., Iofciu, T.: Finding Communities of Practice from User Profiles Based On Folksonomies. In: *Innovative Approaches for Learning and Knowledge Sharing, EC-TEL Workshop Proc.*, pp. 288–297 (2006)
29. Domingos, P., Pazzani, M.J.: On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. *Machine Learning* **29**(2-3), 103–130 (1997)
30. Egozi, O., Gabrilovich, E., Markovitch, S.: Concept-Based Feature Generation and Selection for Information Retrieval. In: D. Fox, C.P. Gomes (eds.) *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008*, pp. 1132–1137. AAAI Press (2008). ISBN 978-1-57735-368-3
31. Eirinaki, M., Vazirgiannis, M., Varlamis, I.: SEWeP: Using Site Semantics and a Taxonomy to enhance the Web Personalization Process. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 99–108. ACM (2003)
32. Fellbaum, C.: *WordNet: An Electronic Lexical Database*. MIT Press (1998)
33. Firan, C.S., Nejdl, W., Paiu, R.: The Benefit of Using Tag-Based Profiles. In: *Proceedings of the Latin American Web Conference*, pp. 32–41. IEEE Computer Society, Washington, DC, USA (2007). DOI <http://dx.doi.org/10.1109/LA-WEB.2007.24>. ISBN 0-7695-3008-7
34. Gabrilovich, E., Markovitch, S.: Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In: *Proceedings of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, pp. 1301–1306. AAAI Press (2006)
35. Gabrilovich, E., Markovitch, S.: Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. In: M.M. Veloso (ed.) *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 1606–1611 (2007)
36. Gemmis, M.d., Lops, P., Semeraro, G., Basile, P.: Integrating Tags in a Semantic Content-based Recommender. In: *Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23–25, 2008*, pp. 163–170 (2008)
37. Giles, J.: Internet Encyclopaedias Go Head to Head. *Nature* **438**, 900–901 (2005)
38. Godoy, D., Amandi, A.: Hybrid Content and Tag-based Profiles for Recommendation in Collaborative Tagging Systems. In: *Proceedings of the 6th Latin American Web Congress (LA-WEB 2008)*, pp. 58–65. IEEE Computer Society (2008). ISBN 978-0-7695-3397-1
39. Goldberg, D., Nichols, D., Oki, B., Terry, D.: Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM* **35**(12), 61–70 (1992). URL <http://www.xerox.com/PARC/dlbx/tapestry-papers/TN44.ps>. Special Issue on Information Filtering
40. Golder, S., Huberman, B.A.: The Structure of Collaborative Tagging Systems. *Journal of Information Science* **32**(2), 198–208 (2006)
41. Gup, T.: Technology and the End of Serendipity. *The Chronicle of Higher Education* (44), 52 (1997)
42. Herlocker, L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems* **22**(1), 5–53 (2004)
43. Holte, R.C., Yan, J.N.Y.: Inferring What a User Is Not Interested in. In: G.I. McCalla (ed.) *Advances in Artificial Intelligence, Lecture Notes in Computer Science*, vol. 1081, pp. 159–171 (1996). ISBN 3-540-61291-2
44. Iaquinta, L., de Gemmis, M., Lops, P., Semeraro, G., Filannino, M., Molino, P.: Introducing Serendipity in a Content-based Recommender System. In: F. Xhafa, F. Herrera, A. Abraham, M. Köppen, J.M. Benitez (eds.) *Proceedings of the Eighth International Conference on Hybrid Intelligent Systems HIS-2008*, pp. 168–173. IEEE Computer Society Press, Los Alamitos, California (2008)

45. Joachims, T., Freitag, D., Mitchell, T.M.: Web Watcher: A Tour Guide for the World Wide Web. In: 15th International Joint Conference on Artificial Intelligence, pp. 770–777 (1997). URL [citeseer.ist.psu.edu/article/joachims97webwatcher.html](http://citeseer.ist.psu.edu/article/joachims97webwatcher.html)
46. Kim, S.B., Han, K.S., Rim, H.C., Myaeng, S.H.: Some Effective Techniques for Naïve Bayes Text Classification. *IEEE Trans. Knowl. Data Eng.* **18**(11), 1457–1466 (2006)
47. Lees-Miller, J., Anderson, F., Hoehn, B., Greiner, R.: Does Wikipedia Information Help Netflix Predictions? In: Seventh International Conference on Machine Learning and Applications (ICMLA), pp. 337–343. IEEE Computer Society (2008). ISBN 978-0-7695-3495-4
48. Lewis, D.D., Ringette, M.: A Comparison of Two Learning Algorithms for Text Categorization. In: Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp. 81–93. Las Vegas, US (1994)
49. Lieberman, H.: Letizia: an Agent that Assists Web Browsing. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 924–929. Morgan Kaufmann (1995)
50. Linden, G., Smith, B., York, J.: Amazon.com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Computing* **7**(1), 76–80 (2003)
51. Magnini, B., Strapparava, C.: Experiments in Word Domain Disambiguation for Parallel Texts. In: Proc. of SIGLEX Workshop on Word Senses and Multi-linguality, Hong-Kong, October 2000. ACL (2000)
52. Magnini, B., Strapparava, C.: Improving User Modelling with Content-based Techniques. In: Proceedings of the 8th International Conference of User Modeling, pp. 74–83. Springer (2001)
53. Mak, H., Koprinska, I., Poon, J.: INTIMATE: A Web-Based Movie Recommender Using Text Categorization. In: Proceedings of the IEEE/WIC International Conference on Web Intelligence, pp. 602–605. IEEE Computer Society (2003). ISBN 0-7695-1932-6
54. McCallum, A., Nigam, K.: A Comparison of Event Models for Naïve Bayes Text Classification. In: Proceedings of the AAAI/ICML-98 Workshop on Learning for Text Categorization, pp. 41–48. AAAI Press (1998)
55. McNee, S.M., Riedl, J., Konstan, J.A.: Accurate is not Always Good: How Accuracy Metrics have hurt Recommender Systems. In: Extended Abstracts of the ACM Conference on Human Factors in Computing Systems (2006)
56. Melville, P., Mooney, R.J., Nagarajan, R.: Content-Boosted Collaborative Filtering for Improved Recommendations. In: Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI-02), pp. 187–192. AAAI Press, Menlo Parc, CA, USA (2002)
57. Michlmayr, E., Cayzer, S.: Learning User Profiles from Tagging Data and Leveraging them for Personal(ized) Information Access. In: Proc. of the Workshop on Tagging and Metadata for Social Information Organization, Int. WWW Conf. (2007)
58. Middleton, S.E., Shadbolt, N.R., De Roure, D.C.: Ontological User Profiling in Recommender Systems. *ACM Transactions on Information Systems* **22**(1), 54–88 (2004)
59. Mihalcea, R., Csomai, A.: Wikify!: Linking Documents to Encyclopedic Knowledge. In: Proceedings of the sixteenth ACM conference on Conference on Information and Knowledge Management, pp. 233–242. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1321440.1321475>. ISBN 978-1-59593-803-9
60. Miller, G.: WordNet: An On-Line Lexical Database. *International Journal of Lexicography* **3**(4) (1990). (Special Issue)
61. Mitchell, T.: *Machine Learning*. McGraw-Hill, New York (1997)
62. Mladenic, D.: Machine learning used by Personal WebWatcher. In: Proceedings of ACAI-99 Workshop on Machine Learning and Intelligent Agents (1999)
63. Mladenic, D.: Text-learning and Related Intelligent Agents: A Survey. *IEEE Intelligent Systems* **14**(4), 44–54 (1999)
64. Montaner, M., Lopez, B., Rosa, J.L.D.L.: A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review* **19**(4), 285–330 (2003)
65. Mooney, R.J., Roy, L.: Content-Based Book Recommending Using Learning for Text Categorization. In: Proceedings of the 5th ACM Conference on Digital Libraries, pp. 195–204. ACM Press, New York, US, San Antonio, US (2000)
66. Moukas, A.: Amalthaea Information Discovery and Filtering Using a Multiagent Evolving Ecosystem. *Applied Artificial Intelligence* **11**(5), 437–457 (1997)
67. Mukherjee, R., Jonsdottir, G., Sen, S., Sarathi, P.: MOVIES2GO: an Online Voting based Movie Recommender System. In: Proceedings of the Fifth International Conference on Autonomous Agents, pp. 114–115. ACM Press (2001)

68. Pazzani, M., Billsus, D.: Learning and Revising User Profiles: The Identification of Interesting Web Sites. *Machine Learning* **27**(3), 313–331 (1997)
69. Pazzani, M.J., Billsus, D.: Content-Based Recommendation Systems. In: P. Brusilovsky, A. Kobsa, W. Nejdl (eds.) *The Adaptive Web, Lecture Notes in Computer Science*, vol. 4321, pp. 325–341 (2007). ISBN 978-3-540-72078-2
70. Pazzani, M.J., Muramatsu, J., Billsus, D.: Syskill and Webert: Identifying Interesting Web Sites. In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, pp. 54–61. AAAI Press / MIT Press, Menlo Park (1996)
71. Picard, R.W.: *Affective Computing*. MIT Press (2000)
72. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In: *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pp. 175–186. ACM, Chapel Hill, North Carolina (1994). URL [citeseer.ist.psu.edu/resnick94grouplens.html](http://citeseer.ist.psu.edu/resnick94grouplens.html)
73. Resnick, P., Varian, H.: Recommender Systems. *Communications of the ACM* **40**(3), 56–58 (1997)
74. Rich, E.: User Modeling via Stereotypes. *Cognitive Science* **3**, 329–354 (1979)
75. Rocchio, J.: Relevance Feedback Information Retrieval. In: G. Salton (ed.) *The SMART retrieval system - experiments in automated document processing*, pp. 313–323. Prentice-Hall, Englewood Cliffs, NJ (1971)
76. Rokach, L., Maimon, O., *Data Mining with Decision Trees: Theory and Applications*, World Scientific Publishing (2008).
77. Sahlgren, M.: *The Word-Space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations between Words in High-dimensional Vector Spaces*. Ph.D. thesis, Stockholm: Stockholm University, Faculty of Humanities, Department of Linguistics (2006)
78. Salter, J., Antonopoulos, N.: CinemaScreen Recommender Agent: Combining collaborative and content-based filtering. *IEEE Intelligent Systems* **21**(1), 35–41 (2006)
79. Salton, G.: *Automatic Text Processing*. Addison-Wesley (1989)
80. Salton, G., McGill, M.: *Introduction to Modern Information Retrieval*. McGraw-Hill, New York (1983)
81. Schwab, I., Kobsa, A., Koychev, I.: Learning User Interests through Positive Examples using Content Analysis and Collaborative Filtering (2001). URL [citeseer.ist.psu.edu/schwab01learning.html](http://citeseer.ist.psu.edu/schwab01learning.html)
82. Sebastiani, F.: Machine Learning in Automated Text Categorization. *ACM Computing Surveys* **34**(1) (2002)
83. Semeraro, G., Basile, P., de Gemmis, M., Lops, P.: User Profiles for Personalizing Digital Libraries. In: Y.L. Theng, S. Foo, D.G.H. Lian, J.C. Na (eds.) *Handbook of Research on Digital Libraries: Design, Development and Impact*, pp. 149–158. IGI Global (2009). ISBN 978-159904879-6
84. Semeraro, G., Degemmis, M., Lops, P., Basile, P.: Combining Learning and Word Sense Disambiguation for Intelligent User Profiling. In: M.M. Veloso (ed.) *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 2856–2861 (2007). ISBN 978-I-57735-298-3
85. Semeraro, G., Lops, P., Basile, P., Gemmis, M.d.: Knowledge Infusion into Content-based Recommender Systems. In: *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, USA, October 22-25, 2009* (2009). To appear
86. Shardanand, U., Maes, P.: Social Information Filtering: Algorithms for Automating “Word of Mouth”. In: *Proceedings of ACM CHI’95 Conference on Human Factors in Computing Systems*, vol. 1, pp. 210–217 (1995). URL [citeseer.ist.psu.edu/shardanand95social.html](http://citeseer.ist.psu.edu/shardanand95social.html)
87. Sheth, B., Maes, P.: Evolving Agents for Personalized Information Filtering. In: *Proceedings of the Ninth Conference on Artificial Intelligence for Applications*, pp. 345–352. IEEE Computer Society Press (1993)
88. Smirnov, A.V., Krizhanovsky, A.: Information Filtering based on Wiki Index Database. *CoRR abs/0804.2354* (2008)
89. Smith, B., Cotter, P.: A Personalized TV Listings Service for the Digital TV Age. *Knowledge-Based Systems* **13**, 53–59 (2000)
90. Sorensen, H., McElligott, M.: PSUN: A Profiling System for Usenet News. In: *Proceedings of CIKM ’95 Intelligent Information Agents Workshop* (1995)
91. Sorensen, H., O’Riordan, A., O’Riordan, C.: Profiling with the INFormer Text Filtering Agent. *Journal of Universal Computer Science* **3**(8), 988–1006 (1997)

92. Stefani, A., Strapparava, C.: Personalizing Access to Web Sites: The SiteIF Project. In: Proc. of second Workshop on Adaptive Hypertext and Hypermedia, Pittsburgh, June 1998 (1998)
93. Straffin, P.D.J.: Topics in the Theory of Voting. The UMAP expository monograph series. Birkhauser (1980)
94. Symeonidis, P.: Content-based Dimensionality Reduction for Recommender Systems. In: C. Preisach, H. Burkhardt, L. Schmidt-Thieme, R. Decker (eds.) Data Analysis, Machine Learning and Applications, Studies in Classification, Data Analysis, and Knowledge Organization, pp. 619–626. Springer Berlin Heidelberg (2008). ISBN 978-3-540-78239-1
95. Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: Tag Recommendations based on Tensor Dimensionality Reduction. In: Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23–25, 2008, pp. 43–50 (2008)
96. Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., Servedio, V.D.P.: Folksonomies, the Semantic Web, and Movie Recommendation. In: Proceedings of the Workshop on Bridging the Gap between Semantic Web and Web 2.0 at the 4th ESWC (2007)
97. Toms, E.: Serendipitous Information Retrieval. In: Proceedings of DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries (2000)
98. Tso-Sutter, K.H.L., Marinho, L.B., Schmidt-Thieme, L.: Tag-aware Recommender Systems by Fusion of Collaborative Filtering Algorithms. In: SAC '08: Proceedings of the 2008 ACM symposium on Applied computing, pp. 1995–1999. ACM (2008). ISBN 978-1-59593-753-7
99. Wasfi, A.M.: Collecting User Access Patterns for Building User Profiles and Collaborative Filtering. In: Proceedings of the International Conference on Intelligent User Interfaces, pp. 57–64 (1999)
100. Witten, I.H., Bell, T.: The Zero-frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression. IEEE Transactions on Information Theory **37**(4) (1991)
101. Yang, Y., Pedersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In: D.H. Fisher (ed.) Proceedings of ICML-97, 14th International Conference on Machine Learning, pp. 412–420. Morgan Kaufmann Publishers, San Francisco, US, Nashville, US (1997). URL [citeseer.ist.psu.edu/yang97comparative.html](http://citeseer.ist.psu.edu/yang97comparative.html)
102. Yeung, C.M.A., Gibbins, N., Shadbolt, N.: A Study of User Profile Generation from Folksonomies. In: Proc. of the Workshop on Social Web and Knowledge Management, WWW Conf. (2008)
103. Zhang, Y., Callan, J., Minka, T.: Novelty and Redundancy Detection in Adaptive Filtering. In: Proceedings of the 25th International ACM SIGIR Conference, pp. 81–88 (2002)
104. Zhao, S., Du, N., Nauerz, A., Zhang, X., Yuan, Q., Fu, R.: Improved Recommendation based on Collaborative Tagging Behaviors. In: Proceedings of International Conference on Intelligent User Interfaces, IUI, pp. 413–416. ACM (2008). ISBN 978-1-59593-987-6



## 基于近邻推荐方法综述

Christian Desrosiers 和 George Karypis

**摘要** 基于近邻(nearest-neighbor)算法广泛用于协同推荐方法中,原因在于该算法简单、有效,且能够提供准确及个性化的推荐。本章将详细综述应用于物品推荐问题的基于近邻算法,特别是该类算法的优势和重要特性。同时本章列出实现这样一个系统时需要做出的基本决策,并给出一些实用经验。本章最后将讨论在大型商业推荐系统中的稀疏性(sparsity)和受限覆盖(limited coverage)的问题,并描述相应的解决方法。

### 4.1 简介

在线购物商城的出现和发展给客户购物方式产生重要影响,这种购物方式能让客户接触到大量的商品和相关的商品信息。随着这种自由购物方式的出现,使得在线商城模式成为亿元行业(即巨大收益行业),同时也使得客户越来越难选择到适合自己的商品。推荐系统是对这种信息过载问题一种重要的解决方法,可以自动提供给客户个性化的商品建议。推荐系统已经广泛应用于众多领域中,例如,书籍、CD[47, 53]、音乐[45, 70]、电影[31, 51, 5]、新闻[6, 41, 76]、幽默笑话[23]和网页[7, 52]。

推荐问题可以定义为评估用户对新物品的反馈,这种评估是基于该系统中历史数据信息,同时推荐那些预测反馈兴趣高的**新颖(novel)**和**独到(original)**的物品给客户。这种基于物品的反馈类型在各个系统中的表现可能不大一样,但大致可以分为三种:**分级反馈(scalar response)**、**二元反馈(binary response)**和**一元反馈(unary response)**。分级反馈也就是人们熟悉的评分制,例如,用数值(1到5星)或比较标准(强烈同意,同意,一般,反对,强烈反对)来表示这些客户对物品的反馈程度。二元反馈仅有两种对立的值来表示评估情况(例如,喜欢/厌恶;感兴趣/不感兴趣)。最后,一元反馈是获取客户与物品互动行为信息(如购买、在线访问等)而没有具体关于用户对于物品的评价。因为大多数用户一般倾向于浏览感兴趣的物品,一元反馈依然能够提供关于客户喜好的有用信息。

获取用户的反馈信息方式也是多种多样的,例如,对于一个电影推荐应用来说,用户可以在观看完电影后给出自己具体的评分,给出相应的观点。用户反馈也可以从具体购买历史或者访问模式中获得[41, 76],例如,当用户花了大量时间浏览具体物品类型的时候,可以认为这是用户感兴趣的物品类型。为了简洁,后面用评分来代替各种类型的用户反馈。

---

Christian Desrosiers, Department of Software Engineering and IT, Ecole de Technologie Supérieure, Montreal, Canada e-mail: christian.desrosiers@etsmtl.ca

George Karypis, Department of Computer Science & Engineering, University of Minnesota, Minneapolis, USA e-mail: karypis@cs.umn.edu

翻译:刘诚 审核:吉林大学-丁彬钊,张彤彤,郑州大学-吴宾

### 4.1.1 问题公式化定义

为了给物品的推荐任务进行公式化定义,我们需要介绍一些基本概念。在系统中我们定义用户集合为 $\mathcal{U}$ ,物品集合 $\mathcal{I}$ 。因此,我们定义 $\mathcal{R}$ 为系统评分集合, $\mathcal{S}$ 为用于评分可选的分数集合(如 $\mathcal{S}=[1, 5]$ 或者 $\mathcal{S}=[\text{喜欢}, \text{不喜欢}]$ )。同时,我们将 $r_{ui}$ 表示为用户 $u \in \mathcal{U}$ 对于特定物品 $i \in \mathcal{I}$ 的评分,同时假定 $r_{ui}$ 的取值个数不能多于一个( $r_{ui}$ 要么有一个取值要么没有取值)。我们用 $\mathcal{U}_i$ 表示集合中已经对物品 $i$ 进行了评分的用户集合。同样, $\mathcal{I}_u$ 表示被用户 $u$ 所评分物品集合。同时被用户 $u$ 和 $v$ 所评分的物品集合( $\mathcal{I}_u \cap \mathcal{I}_v$ )可以表示成 $\mathcal{I}_{uv}$ 。 $\mathcal{U}_{ij}$ 则用于表示同时对物品 $i$ 和物品 $j$ 都进行了评分的用户集合。

**最优项**(best item)和**最优 $N$ 项**(top- $N$ )是推荐系统中最重要两个问题。最优项是指用户 $u$ 最有可能感兴趣的新物品项 $i \in \mathcal{I} \setminus \mathcal{I}_u$ 。当评分值存在时,这个最优项通常可以定义为一个回归或者(多类)分类问题,其目标是用学习函数 $f: \mathcal{U} \times \mathcal{I} \rightarrow \mathcal{S}$ 来预测用户 $u$ 对于物品 $i$ 的评分 $f(u, i)$ 。这样就可以通过这个函数并利用下面的式子预测评价用户 $u$ 对于哪一样物品 $i^*$ 具有最高分数项(也就是最优项):

$$i^* = \underset{i \in \mathcal{I} \setminus \mathcal{I}_u}{\operatorname{argmax}} f(u, i) \quad (4.1)$$

准确性是通常用来评测推荐系统效果的主要方法。一般来说,评分集合 $\mathcal{R}$ 可以分为用于训练函数 $f$ 的训练集 $\mathcal{R}_{\text{train}}$ 和用来测试预测效果的测试集 $\mathcal{R}_{\text{test}}$ 。两种常用的测试预测准确性的标准分别为平均绝对误差(MAE):

$$\text{MAE}(f) = \frac{1}{|\mathcal{R}_{\text{test}}|} \sum_{r_{ui} \in \mathcal{R}_{\text{test}}} |f(u, i) - r_{ui}| \quad (4.2)$$

和均方根误差(RMAE):

$$\text{RMSE}(f) = \sqrt{\frac{1}{|\mathcal{R}_{\text{test}}|} \sum_{r_{ui} \in \mathcal{R}_{\text{test}}} (f(u, i) - r_{ui})^2} \quad (4.3)$$

当没有可用评分信息的时候,例如,仅拥有用户购买商品的列表,评估预测分数的准确性是不可能的。在这种情况下,对于寻找最优项问题通常就转换为向用户推荐感兴趣的列表:列表 $L(u_a)$ 包含用户 $u_a$ 最感兴趣的 $N$ 项物品[18, 45]。评估这种方法的质量也是将物品列表分为用于训练函数 $L$ 的训练集合 $\mathcal{I}_{\text{train}}$ 和用于测试的测试集合 $\mathcal{I}_{\text{test}}$ 。令 $T(u) \subset \mathcal{I}_u \cap \mathcal{I}_{\text{test}}$ 表示测试物品中用户 $u$ 认为相关的物品子集如果用户的反馈是二元反馈(binary responses),那么这些物品项就是指用户 $u$ 所给评分为正的。如果列表中仅给出来用户购买或者浏览了物品项,那么这些物品项就可以直接用于表示 $T(u)$ 。这类方法的效果可通过**准确率**(precision)和**召回率**(recall)进行评估:

$$\text{Precision}(L) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} |L(u) \cap T(u)| / |L(u)| \quad (4.4)$$

$$\text{Recall}(L) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} |L(u) \cap T(u)| / |T(u)| \quad (4.5)$$

这种方法的缺点就是 $L(u)$ 中所有的物品对于用户 $u$ 的感兴趣程度(权重)都是认为等同的。

文献[18]提供了一种解决方法,构造函数 $L$ 可以使得用户 $u$ 所对应的列表 $L(u)$ 所包含的物品是根据用户感兴趣程度排列好的。如果测试集是随机划分的,对每个用户 $u$ ,其对应物品集合 $\mathcal{I}_u$ 中一个项可以表示为 $i_u$ 。那么评估函数 $L$ 的效果可以通过平均逆命中率(Average Reciprocal Hit-Rank, ARHR)衡量:

$$\text{ARHR}(L) = \frac{1}{|U|} \sum_{u \in U} \frac{1}{\text{rank}(i_u L(u))} \quad (4.6)$$

其中  $\text{rank}(i_u L(u))$  表示  $i_u$  在  $L(u)$  中的排名, 如果  $i_u \notin L(u)$ , 那么其值为  $\infty$ 。对于更多关于评估推荐系统效果标准的内容可以阅读本书第 8 章。

## 4.1.2 推荐方法概要

推荐问题真正作为独立研究领域是在 20 世纪 90 年代中期, 推荐系统与其他领域, 如认知科学(cognitive science)[42]和信息检索(information retrieval)[65], 有深厚渊源。解决推荐问题方法通常分为两个大类: 基于内容方法(content-based)和协同过滤方法(collaborative filtering)。

### 4.1.2.1 基于内容方法

基于内容方法[7, 8, 44, 58](基于认知)的大致规则是, 通过用户已经评分的物品来确定这些物品的共同特点, 并利用这些特点为用户  $u$  推荐也有这些特点的新物品。基于内容推荐系统, 可以通过向量  $x_i$  形式来描述物品的大量信息。当物品载体是文本内容时, 如新闻[8, 44]、Web 文档[7, 58], 这些向量就包含了内容有效关键字的“词频-逆文档频率”(Term Frequency-Inverse Document Frequency, TF-IDF)权重[65]。对于用户  $u$ , 对应的偏好属性  $x_u$  通常可以通过物品集合  $I_u$  的内容获得。Rocchio 算法[78, 12]是计算这些偏好的技术, 用于多个基于内容的推荐系统, 如 Newsweeder[44]和 Fab[7]。只要用户  $u$  对物品  $i$  评分, 该技术就会更新用户  $u$  的偏好向量  $X_u$ , 方法是在  $X_u$  里累加  $X_i$  的权重和用户  $u$  对物品  $i$  评分的乘积:

$$x_u = \sum_{i \in I_u} r_{ui} x_i$$

这样就可以利用用户偏好属性集合, 将特征向量  $x_i$  与偏好向量  $x_u$  相似度最接近的物品作为新物品推荐给用户  $u$ 。例如, 利用余弦相似度[7, 8, 44]或者最小描述长度(Minimum Description Length, MDL)[44, 62]方法。这种方法也可以被用来预测用户对新物品的评分[44], 通过为每个评价值  $r \in S$  构建对应内容属性向量  $x_u^{(r)}$  作为用户  $u$  评价过的物品特征向量的平均值。物品  $i$  的预测评分  $\hat{r}_{ui}$  是与  $x_i$  最相似的  $x_u^{(r)}$  的评分值  $r$ 。使用内容信息的贝叶斯算法也提出了预测评价的算法[8, 53, 58]。

推荐系统单纯依赖内容信息通常会受到受限内容分析(limited content analysis)和特化(over-specialization)问题困扰[70]。受限内容分析主要是系统中仅有有限的用户信息或者物品信息。有许多原因造成这种信息的匮乏, 例如, 一些隐私问题影响用户提供个人信息, 或者难以准确地描述物品的内容, 抑或者获取某些项目类型需要代价昂贵, 如音乐、图片。而且, 物品内容也通常难以有效决定其质量, 例如, 在一篇好文章和差文章之间, 如果它们都使用了相同的短语, 那么(文章质量)就不可能区分开。

另一方面, 特化是基于内容推荐系统中推荐新物品时的一个负影响问题, 其中如果一个新物品与用户喜欢已有物品相似, 那么预测用户对该物品的评分则高。例如, 系统给用户推荐的电影, 其风格或者是演员与该用户已经看过的电影一样。解决这类问题方法包括加入随机性[71]或者过滤掉一些过于相似的物品[8, 77]。更多关于内容推荐方法可以查看本书的第 3 章。

### 4.1.2.2 协同过滤方法

基于内容方法是使用被用户评分过的物品内容, 而协同(社会)过滤方法还取决于在系

统中被其他用户评分过的物品内容[18, 31, 41, 47, 60, 45, 70]。其关键点在于, 如果用户  $u$  和用户  $v$  在其他物品评分上很相似, 那么用户  $u$  对新物品  $i$  的评分和另一用户  $v$  对新物品  $i$  的评分也是相似的。同样, 如果其他用户对物品  $i$  和物品  $j$  给了相似的评分, 那么用户  $u$  也会给予这两个物品相似的评分。

协同过滤方法可以克服基于内容方法的一些局限。例如, 当物品的内容不完全或者难以获得的时候, 依然可以通过其他用户的反馈来推荐给用户。同时, 协同过滤推荐是基于用户之间对物品的评价质量, 而不是仅仅依赖可能会干扰质量判断的内容。另外, 与基于内容方法不同的是, 只要其他用户已经显示对这些不同物品的兴趣, 那么协同过滤推荐方法可以推荐内容差异很大的物品。

从文献[1, 5, 10, 18]中可以看到, 协同过滤方法可以大致分为两类: 基于近邻方法和基于模型方法。在基于内存[10]或基于启发式[1])协同过滤方法中[17, 18, 31, 41, 47, 54, 60, 45, 70], 系统中的用户-物品项(评分)可以直接用来对新物品评分。这种基于近邻的方法以通过两种著名的方法来实现, 基于用户或基于物品的推荐。使用了基于用户算法推荐系统, 如 GroupLens[41]、Bellcore video[31]和 Ringo[70], 评估用户  $u$  对物品  $i$  的感兴趣程度, 是通过利用对该物品  $i$  已作评价并且和用户  $u$  有相似评价习惯的其他用户(也叫近邻)。用户  $u$  的近邻是指与用户  $u$  评价很一致的用户, 如在用户  $v$  和用户  $u$  都评价的物品中(即  $\mathcal{I}_{uv}$ ), 用户  $v$  和用户  $u$  评价非常一致。基于物品方法[18, 47, 45]是基于用户  $u$  给相似于  $i$  的物品评分来预测用户  $u$  给物品  $i$  的评分。在这种方法里, 相似物品是指被一些用户评价且具有相似特点的物品。

基于近邻系统是在预测中直接使用已有数据预测, 而基于模型的方法是使用这些评分来学习预测模型。主要思想是使用属性构建用户和物品之间关联, 其属性代表在系统中用户和物品的潜在特征, 如用户喜爱的类别和物品属于的类别。这种模型是通过数据进行训练的, 然后为用户预测新的物品。基于模型方法解决推荐任务有很多, 其中包括贝叶斯聚类(Bayesian Clustering)[10]、潜在语义分析(Latent Semantic Analysis)[32]、潜在狄利克雷分布(Latent Dirichlet Allocation)[9]、最大熵(Maximum Entropy)[78]、玻尔兹曼机(Boltzmann Machines)[64]、支持向量机(Support Vector Machines)[27]和奇异值分解(Singular Value Decomposition)[4, 42, 57, 74, 75]。详细描述基于模型最新进展的方法可以参考本书的第5章。

#### 4.1.3 基于近邻方法的优势

尽管一些最新研究显示最新的基于模型的算法在预测分数方面要好于基于近邻的方法[42, 73], 但是仅预测精度高并不能保证用户得到高效和满意的体验[25]。另一个在推荐系统中影响推荐用户感兴趣物品的因素是惊喜度(serendipity)[25, 45], 即通过帮助用户找到感兴趣但是本不可能发现的物品, 拓展了新颖性概念。例如, 推荐给用户一部他喜欢的导演执导的电影, 如果用户没有意识到要看这部电影, 那么这是一个具有新颖性的推荐。但这不是一个具有惊喜度的推荐, 因为用户很有可能会自己发现这部电影。

基于模型推荐方法在刻画用户爱好的潜在因素方面有突出优势, 例如, 在一个电影推荐系统中, 基于模型方法断定用户是喜剧和浪漫电影的影迷, 而无须准确区分喜剧和浪漫两个维度。这种方法能够推荐给用户一部他不知道的浪漫喜剧。但是, 这种方法难以推荐体裁没有那么高度一致的电影, 例如, 恐怖恶搞喜剧。另一方面, 基于近邻方法能够捕捉这些数据的一些关联, 使用了这种方法的电影推荐系统很有可能推荐给用户与他平常品味

不一样或者不知名的电影，只要他的近邻用户给了这部电影很强的评分。这种推荐系统也许不能保证其成功，就像推荐浪漫喜剧那样，但是它可以帮助用户发现一些新的体裁或者一些新的演员及导演。

基于近邻方法一些优势：

- **简单性**：基于近邻推荐方法直接而且容易实现，在其简单形式里面，仅有一个参数(用于预测的近邻数目)需要调整。
- **合理性**：这种方法对于计算预测也提供了简洁并且直观的理由。例如，在基于物品的推荐系统中，近邻物品列表以及用户给予这些物品的评分，都可以提供给用户作为推荐结果的理由。这能够帮助用户理解推荐结果及其关联性，并且作为交互系统的基础；在该系统中用户能够挑选推荐结果中提示重要性更高的近邻[4]。给用户解释推荐的必要性将在本书的第 15 章阐述。
- **高效性**：选择基于近邻系统的一个强烈原因在于它的效率。大多数基于模型的系统需要大量的时间消耗在训练阶段。由于近邻方法在推荐阶段需要比基于模型方法有更大的消耗，近邻方法可以在离线预先计算近邻，提供近乎即时的推荐结果。同时，存储近邻只需要很小的内存，使得这种方法很适合拥有大量用户或者物品的应用。
- **稳定性**：这种推荐系统拥有另外一个有用的性质就是系统在用户、物品、评分增加时受到影响很小，尤其在大型商业应用中。例如，一旦物品相似计算完成，基于物品的推荐系统就可以为新用户作推荐，而不需要再重新训练系统。同时，一旦新的物品的评分加入，仅仅需要计算该新物品和系统已有物品之间的相似性即可。

#### 4.1.4 目标和概要

本章有两个主要目标，首先可以作为基于近邻推荐系统的指南，描述了如何实现推荐系统的实用信息。特别地，这里将会描述基于近邻推荐方法的主要组成部分和选择使用这些部分的益处。其次，本章描述解决推荐系统一些存在问题的具体方法，如数据稀疏。尽管对于一些简单推荐系统来说，这些方法不是必需的，但是对于各种问题和解决方法有一个广阔的视角，有利于在实现推荐系统时做出合适的决策。

本章结构如下：在 4.2 节中，将介绍基于近邻方法对于预测用户对新物品评分的两种方法：回归和分类，以及它们之间的主要优缺点。本章也会介绍两种实现，基于用户和基于物品的推荐，并且会就推荐系统的准确性、高效性、稳定性、可解释性和惊喜度方面分析这两种实现。4.3 节将描述实现基于近邻推荐系统的主要三个组成部分：评分标准化、相似度权重计算、近邻的选择。描述这三部分的最常用方法以及对应优势比较。在 4.4 节中，将会介绍覆盖率受限和数据稀疏等问题，以及解决这些问题的方法，尤其是基于降维和图的几种技术。最后，将会总结基于近邻推荐的一些特点和方法，在实现这类方法上给予更多的指示。

### 4.2 基于近邻推荐

基于近邻的推荐系统是根据相同“口碑”的准则，即根据和用户自己兴趣相同的人或者根据其他可信源来评价一个物品(电影、书籍、文章、相册等)。为了说明这种方法，我们举一个评分的例子，如图 4.1 所示。



	The Matrix	Titanic	Die Hard	Forrest Gump	Wall-E
John	5	1		2	2
Lucy	1	5	2	5	5
Eric	2	?	3	5	4
Diane	4	3	5	3	

图 4.1 显示四个用户对五部电影评价的小例子

**例 4.1** 用户 Eric 需要决定是否租用他没有看过的电影“Titanic”。他知道 Lucy 在电影上和他有相同的品味，他们都不喜欢“The Matrix”而都喜欢“Forrest Gump”，所以他询问她对于这部电影的观点。另一方面，Eric 发现 Diane 和自己有不同的品味，Diane 喜欢他不喜欢的动作类电影，所以他忽视她的观点，或者考虑和她相反的观点来做出选择。 ◀

#### 4.2.1 基于用户评分

基于近邻用户推荐方法预测用户  $u$  对新物品  $i$  的评分  $r_{ui}$ ，是利用和用户  $u$  兴趣相近且对物品  $i$  作了评分的用户，这些和用户  $u$  兴趣相近的用户称为近邻。假设用  $w_{uv}$  表示用户  $u$  和  $v$  的兴趣相近程度(如何计算这种相近程度会在 4.3.2 节讨论)。用户  $u$  的  $k$  近邻，即  $k$  个与用户  $u$  相似度  $w_{uv}$  最高的用户  $v$  可以表示为  $\mathcal{N}(u)$ 。同时在这些用户中只有对物品  $i$  作了评分的用户才能够用于预测评分  $r_{ui}$ ，所以我们考虑使用  $k$  个和用户  $u$  兴趣相近且对物品  $i$  已作评分的用户代替原来的  $k$  近邻定义，并写作  $\mathcal{N}_i(u)$ 。因此预测评分  $r_{ui}$  可以利用这些近邻对物品  $i$  的平均评分策略获得

$$\hat{r}_{ui} = \frac{1}{|\mathcal{N}_i(u)|} \sum_{v \in \mathcal{N}_i(u)} r_{vi} \quad (4.7)$$

式(4.7)评分公式存在着没有考虑用户  $u$  和每个近邻用户对物品评分相近程度不一的问题。再次回到图 4.1 的例子：假设 Eric 有两个近邻分别是 Lucy 和 Diane，如果对电影“Titanic”评分平均地依靠她们的评分是不合理的，因为 Lucy 的品味和 Eric 更相近。对于此类问题一个通用解决方法就是根据和用户  $u$  的兴趣相近程度进行加权。但是这些权重总和不为 1，这样预测的评分可能会超出评分标准的范围。因此，我们可以标准化这些权重，这样预测评分的准则就变成如下表示：

$$\hat{r}_{ui} = \frac{\sum_{v \in \mathcal{N}_i(u)} w_{uv} r_{vi}}{\sum_{v \in \mathcal{N}_i(u)} |w_{uv}|} \quad (4.8)$$

式(4.8)中的分母，用  $|w_{uv}|$  代替  $w_{uv}$  是因为使用负的权重值会导致预测评分超出允许范围。同样地，可以用  $w_{uv}^\alpha$  代替  $w_{uv}$ ，其中  $\alpha > 0$  是放大因子[10]。就像经常采用的那样，当  $\alpha > 1$  时，与用户  $u$  最接近的用户(评分)就越重要。

**例 4.2** 假设我们使用式(4.8)来预测 Eric 对电影“Titanic”评分，这里会使用到 Lucy 和 Diane 这两个近邻用户对于这部电影评分。进一步假设这些近邻与 Eric 相近权重分别是 0.75 和 0.15。那么这个预测评分为

$$\hat{r} = \frac{0.75 \times 5 + 0.15 \times 3}{0.75 + 0.15} \approx 4.67$$

这个评分更接近 Lucy 而不是 Diane。 ◀

式(4.8)同样有个严重的缺陷：这种方法没有考虑用户会使用不同的评分尺度来衡量

他们对于相同喜欢程度的物品。例如，一个用户也许只会给很少量突出的物品最高分，而另一些用户则会对自己喜欢的物品都给予最高分。解决这个问题可以将近邻评分  $r_{vi}$  进行标准化转换  $h(r_{vi})[10, 60]$ ，可以得到以下预测式子：

$$\hat{r}_{ui} = h^{-1} \left( \frac{\sum_{v \in N_i(u)} w_{uv} h(r_{vi})}{\sum_{v \in N_i(u)} |w_{uv}|} \right) \quad (4.9)$$

需要注意的是这样的预测评分需要转换原始测度，因此将在式子中用到  $h^{-1}$ 。最常用的标准化评分方法将在 4.3.1 节讲述。

#### 4.2.2 基于用户分类

前面所描述的预测方法是通过对近邻用户评分进行加权平均的计算方法，本质上是在解决回归问题。另一方面，基于近邻的分类则是通过用户  $u$  的最近邻对于评分的投票，找出用户  $u$  对物品  $i$  最有可能的评分。用户  $u$  的  $k$  个最近邻对评分  $r(r \in S)$  的投票  $v_{ir}$  可以计为对物品  $i$  评分的所有近邻的相似度权重的总和：

$$v_{ir} = \sum_{v \in N_i(u)} \delta(r_{vi} = r) w_{uv} \quad (4.10)$$

如果  $r_{vi} = r$ ，则  $\delta(r_{vi} = r)$  为 1，否则为 0。如果每个可能的评分值都计算过了，则只要找出  $v_{ir}$  最大的那个  $r$  就是预测出的评分值。

**例 4.3** 再次假设 Eric 的两个近邻 Lucy 和 Diane 的相似权重分别为 0.75 和 0.15。在这个例子中，评分 5 和评分 3 都有一次投票。因为 Lucy 的投票相对于 Diane 的投票有着更大的权重，所以预测评分是  $\hat{r} = 5$ 。

同样可以对考虑了标准化评分的分类方法进行定义：令  $S'$  表示为可能的已经标准化的值的集合(可能需要经过离散化处理)，预测分数可以定义为

$$\hat{r}_{ui} = h^{-1} \left( \operatorname{argmax}_{r \in S'} \sum_{v \in N_i(u)} \delta(h(r_{vi}) = r) w_{uv} \right) \quad (4.11)$$

#### 4.2.3 回归与分类

选择基于近邻回归或者分类方法很大程度取决于系统的评分刻度(scale)类型。因此，如果一个评分刻度是连续的，如 Jester 笑话推荐系统[23]中的评分可以在 -10 到 10 之间取任意值，那么回归方法更加适合。相反，如果评分刻度仅是一些离散的值，如“好”“差”，抑或者是数值没有明显排序，那么分类方法应该会更加适合。另外，因为标准化方法会使评分映射到连续类型，这就导致分类方法很难处理这类问题。

还有一种办法可以比较这两种方法，就是假设所有近邻的相似度权重都相同。当用于预测的近邻数增加时，回归方法预测的分数  $r_{ui}$  将趋向于物品  $i$  的评分的平均数。假设物品  $i$  的评分仅在评分范围的极端，如喜欢或者厌恶，那么回归方法将会是一个保险的决策，使得物品的分值较为平均。从统计观点出发，这也是合理的，因为期望评分(也就是预测评分)是能最小化 RMSE 的分值。另一方面，分类方法将会用对物品  $i$  最频繁的评分作为预测评分，这也是最冒险方法，因为物品要么标示成“好”或者要么标示“坏”。但是，正如前面所提到的，选择冒险的方法也是非常具有吸引力的，因为这样会产生惊喜度的推荐。

#### 4.2.4 基于物品推荐

基于用户的推荐方法是依赖于和自己兴趣相同的用户来预测一个评分,而基于物品的推荐方法[18, 47, 45]是通过评分相近的物品来预测。让我们通过以下例子来说明这种方法。

**例 4.4** 与咨询同伴不同, Eric 通过他已经看过的电影来决定电影“Titanic”是否适合他。他发现对这部电影评分的人对于电影“Forrest Gump”和“Wall-E”给予相近的评分。因为 Eric 也喜欢这两部电影, 所以他认为自己也会喜欢“Titanic”。

这种方法描述如下: 定义  $\mathcal{N}_u(i)$  为用户  $u$  已经评分且和物品  $i$  评分相近的物品。用户  $u$  对物品  $i$  的预测评分可以通过对用户  $\mathcal{N}_u(i)$  中物品的评分进行加权平均运算:

$$\hat{r}_{ui} = \frac{\sum_{j \in \mathcal{N}_u(i)} w_{ij} r_{uj}}{\sum_{j \in \mathcal{N}_u(i)} |w_{ij}|} \quad (4.12)$$

**例 4.5** 假设我们的预测还是用两个近邻, “Titanic”的相似项“Forrest Gump”和“Wall-E”的相似权重分别为 0.85 和 0.75。因为 Eric 对这两部电影评分分别是 5 和 4, 因此预测评分可以计算为

$$\hat{r} = \frac{0.85 \times 5 + 0.75 \times 4}{0.85 + 0.75} \approx 4.53$$

不同用户有自己的独立评分尺度, 所以考虑使用  $h$  对评分标准化:

$$\hat{r}_{ui} = h^{-1} \left( \frac{\sum_{j \in \mathcal{N}_u(i)} w_{ij} h(r_{uj})}{\sum_{j \in \mathcal{N}_u(i)} |w_{ij}|} \right) \quad (4.13)$$

此外, 我们也可以定义基于物品分类的方法。在这种情况下, 用户  $u$  评分的物品  $j$  会决定另一个物品  $i$  的评分, 而且这种决定权会根据  $i$  和  $j$  之间的相似度加权。这个方法的标准版本如下:

$$\hat{r}_{ui} = h^{-1} \left( \operatorname{argmax}_{r \in \mathcal{S}^i} \sum_{j \in \mathcal{N}_u(i)} \delta(h(r_{uj}) = r) w_{ij} \right) \quad (4.14)$$

#### 4.2.5 基于用户和基于物品推荐的对比

当需要选择是基于用户还是基于物品近邻方法来实现推荐系统的时候, 有五个准则需要考虑:

- **准确性:** 推荐系统的准确度很大程度依赖于系统中用户数与物品数之间的比例。如 4.3.2 节描述, 基于用户方法决定用户的近邻数的相似度计算是通过比较这些用户对相同物品评分。假设一个系统有 1000 个用户对 100 个物品的 10 000 个评分, 为了便于分析, 假设这些评分是标准分布的<sup>①</sup>。从表 4.1 可以得到平均有 650 个用户可以成为潜在近邻。然而用于计算相似度的共同评分数量仅为 1。另一方面, 基于物品方法计算相似度比较同一个用户对两个不同物品评分。再次假设这些评分是符合标准分布的, 我们可以发现潜在近邻的平均个数是 99, 可以用于计

① 真实数据中评分分布通常是偏态的, 也就是说, 大多数评分都落在一小部分物品上。

算相似度的评分平均个数是 10。

表 4.1 基于用户和基于物品近邻方法中分别用于计算相似性的近邻平均数量和评分平均数量。假设评分是正态分布的, 每个用户的平均评分数是  $p = |\mathcal{R}|/|u|$ , 每个物品的平均评分数是  $q = |\mathcal{R}|/|i|$

	近邻平均数量	评分平均数量
基于用户	$( u -1)\left(1-\left(\frac{ i -p}{ i }\right)^p\right)$	$\frac{p^2}{ i }$
基于物品	$( i -1)\left(1-\left(\frac{ u -q}{ u }\right)^q\right)$	$\frac{q^2}{ u }$

通常, 一小部分高可信度的用户要比一大部分相似度不是那么可信的近邻要适合得多。对于用户数量远远大于物品数量的大型商业系统, 如 Amazon.com, 基于物品的推荐方法更加准确[19, 45]。同样, 用户数少于物品数的系统, 如科研论文推荐系统只有几千名用户, 但却有成千上万的论文要推荐, 可能采用基于用户的近邻方法会更有益[25]。

- **效率:** 如表 4.2 所示, 推荐系统的内存和计算效率也依赖用户数量和物品数量的比例。因此当用户数量远远大于物品数量时, 基于物品推荐方法在(训练阶段)计算相似度权重方面所需的内存和时间要远远小于基于用户的方法。但是, 在线推荐阶段的时间复杂度因为只依赖有效的物品数和近邻数的最大值, 所以对于基于用户和基于物品方法来说是相同的。

表 4.2 基于用户和基于物品的近邻推荐方法的空间复杂度及时间复杂度, 其函数包括三个变量, 分别是为每个用户的评分数最大值  $p = \max_u |\mathcal{R}_u|$ 、每个物品的评分数最大值  $q = \max_i |\mathcal{R}_i|$  和评分预测中用到的近邻个数最大值  $k$

	空间复杂度	时间复杂度	
		训练	在线
基于用户	$O( u ^2)$	$O( u ^2 p)$	$O( i  k)$
基于物品	$O( i ^2)$	$O( i ^2 q)$	$O( i  k)$

在实际中, 计算相似权重要远远小于表 4.2 所描述最坏情况的计算复杂度, 原因在于用户仅有效评价少量的物品。根据这种情况, 仅那些非零的相似权重需要被存储下来, 这样就远远小于用户之间成对的数量。这些存储数量可以进一步通过存储最大  $N$  权重( $N$  为参数)来减少数量[45]。同样地, 不用测试每对用户或物品可以有效地计算非零权重, 这使得近邻方法对非常大型的系统更具有可扩展性。

- **稳定性:** 选择基于用户或者基于物品方法也依赖用户或者物品的改变频率和数量。如果系统中有效物品的列表比用户相对稳定, 那么基于物品的推荐方法可能更适用, 因为物品相似度依然能够用于推荐物品给新的用户。相反, 当应用物品的列表经常改变, 如文字推荐系统, 基于用户的方法会更加稳定。
- **合理性:** 基于物品的推荐方法的优点是易于证明推荐的合理性。因此, 预测中用到的近邻物品列表, 以及它们的相似度权重, 都可以作为推荐结果的解释提供给用户。通过修改近邻列表及其权重, 使用户在推荐过程中参与交互成为可能。但是, 基于用户的方法就很难做到这点, 因为活跃用户不知道在推荐结果中起到近邻作用的其他用户。

- **惊喜度**：对于基于物品推荐方法来说，对于物品评分预测是基于评分相似的物品。这样，推荐系统推荐一个物品给用户，这个物品通常已经用户喜欢的类型。例如，在电影推荐应用中，具有相同体裁、演员或者导演的电影会非常可能推荐给用户。这种方法可以产生安全的推荐，但是这种方法难以帮助用户发现他也可能非常喜欢的其他不同类型物品。

另一方面，因为是根据用户相似度来进行的，基于用户的方法可以产生更加新颖的推荐。当推荐是基于很小部分近邻数的时候尤为有效。例如，用户  $A$  仅因为几部喜剧电影和用户  $B$  相似，但是用户  $B$  又喜欢不同体裁的电影，这样也会有可能因为  $A$  和  $B$  相似而把这类电影推荐给  $A$ 。

### 4.3 近邻方法的要素

在前面几节已经看到，选择回归或者分类，以及选择基于用户还是基于物品的方法，都会对推荐系统的准确性、效率和整体质量产生重要影响。除了这些重要的属性，在推荐系统的实现中还有三个非常重要的因素需要考虑：1) 评分标准化；2) 相似度权重的计算；3) 近邻的选择。下面介绍一些关于这三类问题的通用方法，描述主要的优缺点，以及如何实现它们。

#### 4.3.1 评分标准化

当一个用户对一个物品给予评分的时候，每个用户都有自己的评价准则。即使显式地定义每个评分的意义(如 1 表示强烈不同意，2 表示不同意，3 表示中立等)，有些用户依然会不愿意给他们喜欢的物品评高分或给他们不喜欢的物品评低分。均值中心化(mean-centering)和 Z-score 这两种通用的标准化机制可以将个人评分标准转换到更一般的整体评分标准。

##### 4.3.1.1 均值中心化

均值中心化方法[10, 60]的思想就是通过与平均分的比较来决定一个评分为正或者为负。在基于用户推荐系统中，设  $r_{ui}$  为用户对物品  $i$  的原始评分，可以通过减去他评价的所有物品的平均分  $\bar{r}_u$ ，转化为均值中心化评分：

$$h(r_{ui}) = r_{ui} - \bar{r}_u$$

可以用下式来预测用户评分  $r_{ui}$ ：

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N_i(u)} w_{uv} (r_{vi} - \bar{r}_v)}{\sum_{v \in N_i(u)} |w_{uv}|} \quad (4.15)$$

同样，基于物品来说， $r_{ui}$  的均值中心化评分也可以这样得

$$h(r_{ui}) = r_{ui} - \bar{r}_i$$

其中， $\bar{r}_i$  表示用户集合  $U_i$  对物品  $i$  的平均评分。这种标准化技术通常用在基于物品推荐中，其中  $r_{ui}$  可以预测为

$$\hat{r}_{ui} = \bar{r}_i + \frac{\sum_{j \in N_u(i)} w_{ij} (r_{uj} - \bar{r}_j)}{\sum_{j \in N_u(i)} |w_{ij}|} \quad (4.16)$$

中心化均值方法有个有趣的性质就是：用户对物品喜好倾向可以直接观察标准化后的



评分值的正负情况。同时评分可以表示用户对该物品喜好或厌恶的程度。

**例 4.6** 如图 4.2 所示，尽管 Diane 给予电影“Titanic”和“Forrest Gump”的评价分数为 3 分，但是用户的平均中心化评分结果却显示其对于这些电影的偏好是负值。这是因为她的评分总体上高于所有用户的整体平均分，所以整体的平均分对她来说表示不感兴趣。两种均值中心化方法的区别也可以对比出来。例如，Diane 对“Titanic”的物品均值中心化评分是中立而非否定的，原因在于大多数用户对这部电影的评分都较低。同样地，在物品均值中心化评分中，Diane 对于电影“The Matrix”的喜欢程度以及 John 对于“Forrest Gump”的不喜欢程度要更加强烈。

用户均值中心化					
	The Matrix	Titanic	Die Hard	Forrest Gump	Wall-E
John	2.50	-1.50		-0.50	-0.50
Lucy	-2.60	1.40	-1.60	1.40	1.40
Eric	-1.50		0.50	1.50	0.5
Diane	0.25	-0.75	1.25	-0.75	

物品均值中心化					
	The Matrix	Titanic	Die Hard	Forrest Gump	Wall-E
John	2.00	-2.00		-1.75	-1.67
Lucy	-2.00	2.00	-1.33	1.25	1.33
Eric	-1.00		-0.33	1.25	0.33
Diane	1.00	0.00	1.67	-0.75	

图 4.2 用户和物品的均值中心化评分

4.3.1.2 Z-score 标准化

考虑这种情况：用户 A 和用户 B 平均评分都是 3，但是假设用户 A 的评分在 1 到 5 之间轮流选择，而用户 B 则都是 3。如果用户 B 给物品评 5 分，这会比用户 A 给物品评 5 分更加意外，因此反映了用户 B 更加喜爱这个物品。均值中心化方法移除了针对平均评分的不同感受而导致的偏差，而 Z-score 标准化[29]方法则还考虑了个人评分范围不同带来的差异性。同样，这种方法在基于用户方法和基于物品方法之间也有差异。在基于用户方法中，标准化评分等于用户均值中心化评分除以用户评分标准差  $\sigma_u$ ：

$$h(r_{ui}) = \frac{r_{ui} - \bar{r}_u}{\sigma_u}$$

因此，基于用户的预测评分  $r_{ui}$  可以通过下式计算：

$$\hat{r}_{ui} = \bar{r}_u + \sigma_u \frac{\sum_{v \in N_i(u)} w_{uv}(r_{vi} - \bar{r}_v) / \sigma_v}{\sum_{v \in N_i(u)} |w_{uv}|} \tag{4.17}$$

同样地，基于物品方法使用 Z-score 标准化  $r_{ui}$  是通过物品均值中心化评分除以物品评分的标准差得到：

$$h(r_{ui}) = \frac{r_{ui} - \bar{r}_i}{\sigma_i}$$

因此，基于物品的预测评分  $r_{ui}$  可以通过下式计算：

$$\hat{r}_{ui} = \bar{r}_i + \sigma_i \frac{\sum_{j \in N_u(i)} w_{ij} (r_{uj} - \bar{r}_j) / \sigma_j}{\sum_{j \in N_u(i)} |w_{ij}|} \quad (4.18)$$

#### 4.3.1.3 选择一个标准化方法

对一些例子, 评分标准化可能会产生意料之外的效果。例如, 假设一个用户只给他买过的物品打最高分, 那么平均中心化方法会认为这个用户“容易满足”, 那么所有低于最高评分的(不管是正的还是负的评分)都会被考虑为负分。但是这个用户事实上可能是“难以伺候”并谨慎地选择一些他确定喜欢的物品。所以, 对少量评分进行标准化可能会产生不可预测的结果。例如, 如果一个用户只是给予一个评分或者几个相同的评分, 那么他评分的标准差将会是 0, 这就导致一个不可定义的预测值。尽管如此, 只要评分数据不是极度稀疏的, 标准化评分的方法还是可以改进预测的[29, 33]。

对比均值中心化和 Z-score 方法, 如前面描述, 后者因为考虑了基于用户或物品评分的方差而具有额外优势。特别是, 在处理范围很大的离散评分或者连续值评分时尤为有用。另一方面, 因为 Z-score 方法除以了评分的标准差值, 所以它比均值中心化方法更加敏感, 用它预测的评分经常会超过评分范围。最后, 尽管早先研究[29]中的结论是均值中心化和 Z-score 方法的结果相似, 更新的研究[33]显示 Z-score 方法还是具有额外优势的。

如果标准化方法不太可能或者不能改进结果, 另一种可行的解决方法就是基于偏好的过滤(preference-based filtering)方法。相对于关注评分绝对值, 这种方法更关注预测用户的相对偏好, 因为评分的范围不改变对物品的偏好次序, 所以对相对偏好的预测并不需要标准化评分。以下文献介绍了这种方法[13, 21, 37, 36]。

#### 4.3.2 相似度权重计算

相似度权重在基于近邻的推荐方法中扮演双重角色: 1) 可以用于选择可信的近邻用于预测评分; 2) 给予不同近邻在预测中的权重。计算相似度权重是基于近邻推荐系统中最重要的一个方面, 它可以直接影响准确性和性能。

##### 4.3.2.1 基于关联的相似度

在信息检索中经常这样计算对象  $a, b$  之间的相似度: 首先将  $a$  和  $b$  表示成向量形式  $(\mathbf{x}_a, \mathbf{x}_b)$ , 然后计算两向量间的余弦向量(或向量空间)相似度:

$$\cos(\mathbf{x}_a, \mathbf{x}_b) = \frac{\mathbf{x}_a^T \mathbf{x}_b}{\|\mathbf{x}_a\| \|\mathbf{x}_b\|}$$

在基于物品的推荐系统中, 这种方法改为计算用户的相似性, 将用户  $u$  表示为一个向量  $\mathbf{x}_u \in \mathbf{R}^{|I|}$ , 其中  $x_{ui} = r_{ui}$  为用户  $u$  对物品  $i$  的评分, 0 表示没有评分。那么用户  $u$  和  $v$  之间的相似度可以这样计算:

$$CV(u, v) = \cos(\mathbf{x}_u, \mathbf{x}_v) = \frac{\sum_{i \in I_{uv}} r_{ui} r_{vi}}{\sqrt{\sum_{i \in I_u} r_{ui}^2 \sum_{j \in I_v} r_{vj}^2}} \quad (4.19)$$

其中  $I_{uv}$  表示同时被用户  $u$  和  $v$  都评分的物品。这种方法存在一个问题在于它没考虑用户  $u$  和  $v$  的评分均值以及方差间的差异。

一个常用的可以除去均值和方差间差异影响的方法是皮尔逊相似度(Pearson Correla-

tion, PC):

$$PC(u,v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)^2 \sum_{i \in I_{uv}} (r_{vi} - \bar{r}_v)^2}}$$

(4.20)

注意这和先进行 Z-score 标准化评分, 然后计算余弦相似度是不同的: 皮尔逊相似度仅考虑了用户评分交集的标准差, 而不是全部。同样的方法可以用于计算物品  $i$  和  $j$  的相似度[18, 45], 基于评分用户的交集:

$$PC(i,j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)^2 \sum_{u \in U_{ij}} (r_{uj} - \bar{r}_j)^2}}$$

(4.21)

其中正负号表示关联的同向或相反性, 其值表示关联的强度。

**例 4.7** 图 4.3 给出了用户和物品间皮尔逊相似度的计算: 我们可以看到, Lucy 的电影品味和 Eric 很接近(相似度为 0.922), 但与 John 则非常不同(相似度为 -0.938)。这就意味着 Eric 的评分可以用来预测 Lucy 的评分, 但是 Lucy 与 John 的评分不相关或者干脆相反。我们也同时可以发现喜欢电影“The Matrix”的人同时喜欢“Die Hard”但是厌恶“Wall-E”。值得注意的是, 这些关联的发现并没依赖任何关于电影体裁、导演和演员的信息。

基于用户的皮尔逊相似度				
	John	Lucy	Eric	Diane
John	1.000	-0.938	-0.839	0.659
Lucy	-0.938	1.000	0.922	-0.787
Eric	-0.839	0.922	1.000	-0.659
Diane	0.659	-0.787	-0.659	1.000

基于物品的皮尔逊相似度					
	The Matrix	Titanic	Die Hard	Forrest Gump	Wall-E
Matrix	1.000	-0.943	0.882	-0.974	-0.977
Titanic	-0.943	1.000	-0.625	0.931	0.994
Die Hard	0.882	-0.625	1.000	-0.804	-1.000
Forrest Gump	-0.974	0.931	-0.804	1.000	0.930
Wall-E	-0.977	0.994	-1.000	0.930	1.000

图 4.3 用户和物品 PC 相似性评分

用户间评分的方差要明显大于物品间的评分, 因此计算物品间相似度时, 相比于用物品均值中心化, 用户均值中心化更加适合。调整的余弦相似度(Adjusted Cosine, AC)就是这样:

$$AC(i,j) = \frac{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_u)^2 \sum_{u \in U_{ij}} (r_{uj} - \bar{r}_u)^2}}$$

在一些基于物品的例子中, 用调整的余弦相似度算法所作的预测要好于皮尔逊相似度算法[45]。

4.3.2.2 其他的相似度权重计算方法

下面是一些其他的相似度算法: 均方差(Mean Squared Difference, MSD)[70]是使用

用户  $u$  和  $v$  对相同物品评分差的平方总和均值的倒数表示两个人的相似度：

$$\text{MSD}(u, v) = \frac{|\mathcal{I}_{uv}|}{\sum_{i \in \mathcal{I}_{uv}} (r_{ui} - r_{vi})^2} \quad (4.22)$$

尽管这种方法经过修改可用来计算两组标准化评分的差异，与皮尔逊相似度相比，MSD 算法不能表示用户偏好的负关联，或者对不同物品的喜好程度。而包含这种负关联可能提高预测的准确度[28]。

另一种有名的相似度算法是斯皮尔曼等级关联(Spearman Rank Correlation, SRC)[39]。和皮尔逊方法直接运用评分值不同，斯皮尔曼等级关联考虑运用这些评分的排名。定义  $k_{ui}$  为物品  $i$  在用户  $u$  所评分物品中的排位(并列评分用它们的平均排名)，则用户  $u$  和  $v$  的相似度可以这样计算：

$$\text{SRC}(u, v) = \frac{\sum_{i \in \mathcal{I}_{uv}} (k_{ui} - \bar{k}_u)(k_{vi} - \bar{k}_v)}{\sqrt{\sum_{i \in \mathcal{I}_{uv}} (k_{ui} - \bar{k}_u)^2 \sum_{i \in \mathcal{I}_{uv}} (k_{vi} - \bar{k}_v)^2}} \quad (4.23)$$

其中， $\bar{k}_u$  是用户所评价物品的平均排名。

斯皮尔曼等级关联算法的主要优势在于排名可以绕开标准化评分的问题。而当用户评分只有少量可选值的时候，这种方法不是最好的，原因在于它会产生大量的并列排名。同时由于需要排序，它的消耗要明显大于皮尔逊关联算法。

表 4.3 显示了 MSD、SRC 和 PC 方法在 MovieLens<sup>⊖</sup>数据集[28]上基于用户预测的准确度(MAE)，给出了不同  $k$  值的结果( $k$  表示预测时取的最大近邻数)。我们发现可能是由于没有考虑到负关联，MSD 得到的预测准确度是最低的。同时发现 PC 方法比 SRC 方法准确度要高一点。尽管 PC 方法通常被认为是最优的相似度计算方法[28]，但是最近一些研究显示，这些算法的效果很大程度依赖于数据本身[33]。

表 4.3 随着近邻  $k$  数目的预测结果

$k$	MSD	SRC	PC
5	0.789 8	0.785 5	0.782 9
10	0.771 8	0.763 6	0.761 8
20	0.763 4	0.755 8	0.754 5
60	0.760 2	0.752 9	0.751 8
80	0.760 5	0.753 1	0.752 3
100	0.761 0	0.753 3	0.752 8

#### 4.3.2.3 关于重要性的说明

在系统中，由于评分数据相对于用户数或物品数通常是稀疏的，相似度也仅是通过很少一部分对相同物品的评分或者同一用户所做的评分来计算获得。例如，系统中有 1000 个用户对 100 个物品的 10 000 个评分(假设满足均匀分布)。表 4.1 是通过比较他们共同评价的一个物品而计算出的两个用户的相似度。如果这几个评分都是相等的，那么这些用户则被视为“完全相似”并且在互相推荐中起到非常重要作用。但是如果这些用户的爱好事实上是不同的，就会导致一个不好的推荐。

⊖ <http://www.grouplens.org/>

人们提出了一些考虑加入相似度权重重要性(significance)的策略,这些策略的本质是一样的:当只有少量评分用于计算时,就会降低相似度重要性的权重。

例如,在重要性权重(Significance weighting)里[29, 49],当两人共同评分的物品数  $I_{uv}$  小于给定的参数  $\gamma > 0$  时,他们的相似度  $w_{uv}$  会受到与  $I_{uv}$  成比例的惩罚,公式如下:

$$w'_{uv} = \frac{\min\{|I_{uv}|, \gamma\}}{\gamma} \times w_{uv} \quad (4.24)$$

同样,对于共同评分用户过少的物品间相似度  $W_{ij}$ ,也可以这样调整为

$$w'_{ij} = \frac{\min\{|U_{ij}|, \gamma\}}{\gamma} \times w_{ij} \quad (4.25)$$

在文献中[29, 28]发现  $\gamma \geq 25$  可以显著地提高预测评分的准确性,其中  $\gamma$  为 50 的时候可以获得最好的结果。由于最优参数值是依赖于数据本身的,应该使用交叉验证方法来决定。

重要性权重方法的一个特点就是,当一个权重需要调整的时候需要使用阈值  $\gamma$  来决定。文献[4]描述了一个更加连续的方法,这种方法基于“收缩”概念。这种方法使用贝叶斯观点来解释,一个参数的最优估计是后验均值,对应参数的先验均值(空值)和完全基于数据的经验估计量的线性组合。在这种情况下,要预估的参数就是相似度权重,空值为零。因此一个在少量评分情况下的用户相似性权重估计可以这样表示:

$$w'_{uv} = \frac{|I_{uv}|}{|I_{uv}| + \beta} \times w_{uv} \quad (4.26)$$

其中,  $\beta > 0$  是一个需要交叉验证选择的参数。在这种方法里,  $w_{uv}$  按比例收缩为  $\beta/|I_{uv}|$ , 因此当  $|I_{uv}| \gg \beta$  的时候,调整几乎没有改变。同样,物品的相似性收缩为

$$w'_{ij} = \frac{|U_{ij}|}{|U_{ij}| + \beta} \times w_{ij} \quad (4.27)$$

如文献[4]所说,  $\beta$  的值通常为 100。

#### 4.3.2.4 关于差异的说明

两个用户对物品给出一致的喜欢或不喜欢评分,可能会不如他们给出差异较大的评分时提供更多的信息量。例如,多数人都喜欢经典电影“The Godfather”,所以这部电影的权重通常会是一个很高的值。同样,一个用户对于物品以同样方式给予评分所提供的信息,要远远低于他对不同物品体现出不同偏好的变化所显示的信息。

推荐方法解决这类问题是使用反用户频率(Inverse User Frequency)[10]。基于信息检索领域反文档频率(Inverse Document Frequency, IDF)概念,每个物品  $i$  都会赋以权重  $\lambda_i$ , 对应评论了物品  $i$  的用户比例的  $\log$  值:

$$\lambda_i = \log \frac{|U|}{|U_i|}$$

当计算用户  $u$  和  $v$  的频率加权皮尔逊关联系数(Frequency-Weighted Pearson Correlation, FWPC),对于物品  $i$  的评分间关联性需要给予权重  $\lambda_i$ :

$$FWPC(u, v) = \frac{\sum_{i \in I_{uv}} \lambda_i (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} \lambda_i (r_{ui} - \bar{r}_u)^2 \sum_{i \in I_{uv}} \lambda_i (r_{vi} - \bar{r}_v)^2}} \quad (4.28)$$

这种方法可以改进基于用户推荐方法的预测准确性[10],也可以适用于物品相似性计算。

研究人员还提出了更多考虑了评分差异的高级策略。文献[35]描述了其中一种方法是



通过最大化用户之间平均相似性来计算因子  $\lambda_i$ 。在这种方法中, 给定物品的权重向量为  $\lambda = (\lambda_1, \dots, \lambda_T)$ , 用户  $u$  和  $v$  之间的相似度可以估算为用户  $u$  与用户  $v$  有着相同评分的可能性:

$$\Pr(u | v, \lambda) = \frac{1}{Z_v} \exp\left(\sum_{i \in I_{uv}} \lambda_i r_{ui} r_{vi}\right)$$

其中  $Z_v$  是一个标准化常量, 最优物品权重向量是能够最大化用户间平均相似性的那个。

### 4.3.3 近邻的选择

近邻数量的选择和选择近邻规则对于推荐系统的质量会产生重要影响。选择用于推荐系统的近邻方法通常可以分为两步: 1) 用全局过滤步骤保持最有可能的近邻; 2) 在预测每一步中选择最合适的近邻作预测。

#### 4.3.3.1 过滤预选近邻数

在大型的推荐系统中可能会包含几百万条用户和物品信息, 由于内存限制, 它不太可能存储所有用户间和物品间的非零相似度。这样做也是极大浪费, 因为只是一小部分有重要影响的值是用来预测的。预选近邻数是一项基本步骤, 它通过减少存储相似度权重数量, 并且限制用于预测的近邻数目, 使得基于近邻推荐方法可行。这里有几种方法可以实现:

- **Top-N 过滤:** 对于每一个用户或者物品来说, 列表中仅仅有  $N$  个近邻和对应的相似度权重是需要保留的。为了防止带来的效率和准确性问题,  $N$  的选择需要比较谨慎。因此, 当  $N$  选择太大的时候, 需要大量的内存来存储近邻列表, 同时预测速度也会变得很慢。另一方面,  $N$  选择得太小则会减小推荐方法的覆盖, 这样会使得一些物品永远推荐不出来。
- **阈值过滤:** 与选择固定数量的近邻数目不同, 这种方法保留所有相似度权重数大于一个给定的阈值  $w_{\min}$  的近邻。对比前面方法, 尽管这种方法更加灵活, 因为仅仅最重要的近邻会被保留, 但同时阈值  $w_{\min}$  的选择也是比较困难的。
- **负值过滤:** 通常负评分关联比正关联可靠性要差。显而易见地, 这是因为两个用户间很强的正关联性预示他们是属于同一团体的(例如, 青少年、科幻迷等)。但是, 尽管负关联性可能显示其属于不同团体, 但它不能显示这些团体差异程度, 或者这些团体是否可以兼容其他类型的物品。一些实验性的研究[29, 25]发现负关联性对预测准确性没有显著的提高, 根据数据可以决定是否需要丢弃这些关联关系。

值得注意的是, 这三种过滤方法并不是互斥的, 并且可以为了推荐系统的需要结合在一起。例如, 可以在丢弃所有负向相似度的同时也丢弃那些相似度比给定阈值小的关联关系。

#### 4.3.3.2 用于预测的近邻

一旦每个用户或者物品的候选近邻列表被计算出来, 对一个新的评分预测可以通过  $k$  近邻方法得到,  $k$  近邻也就是相似权重最大的  $k$  个近邻。现在最重要的问题在于如何选择  $k$  的值。

表 4.3 显示, 通过观察所得的预测准确度随着  $k$  值的增加通常呈现为一个凹的函数。因此当近邻数目限制在一个很小的数的时候(如  $k < 20$ ), 预测精度通常会很低。当  $k$  增加时, 越来越多的近邻参与到预测中, 那么那些由单独近邻所造成的偏差就会被平均掉。这

样的结果则使预测精度得到提高。最后,由于很多近邻加入进来进行预测,准确度通常会下降(如 $k > 50$ ),这样原因在于一些重要的关联被一些不重要的关联所削弱。尽管在一些文献中,近邻数目大概在20~50[28, 25],但最优的 $k$ 值依然需要通过交叉验证来实现。

最后请注意,基于少量非常相似用户可能得到更加新颖的推荐结果,但代价是降低准确度。例如,推荐系统可以找到该用户最相似的用户,推荐的新物品则是这个最相似用户的最高评分物品。

## 4.4 高级进阶技术

基于近邻的方法是基于评分之间关联性的,如前面所提及的一些方法,这类方法有两个重要的缺陷:

- **覆盖受限:** 由于计算两个用户间的相似性是基于他们对相同物品的评分,而且只有对相同物品进行了评分的用户可以作为近邻。这些假设是非常受限制的,例如,有些用户有很少或者没有共同评分但依然有相似的兴趣。而且,仅仅被近邻用户评价过的物品才会被推荐,推荐方法的覆盖将受到限制。
- **对稀疏数据的敏感:** 4.2.5节所简单提到的另一个评分之间关联性问题就是,基于近邻推荐方法的准确性会受到缺少有效评分限制。由于用户通常只是对一小部分物品进行了评分[7, 25, 68, 67],稀疏性是大多数推荐系统共同问题。新加入系统的用户或者物品根本没有评分,也就是所谓的“冷启动”问题[69],所以这种情况下,问题更为严重。当数据是稀疏的时候,两个用户或物品之间就很难有相同的评分,这就造成基于近邻方法在预测评分的时候仅使用了很少量有限的近邻。另外,相似性权重的计算也可能依赖小部分的评分,从而导致推荐偏差(此类问题可以查看4.3.2.3节)。

解决这类问题一个常用方法就是用默认的数据去填补缺失的评分[10, 18],例如,评分范围的中值,或者用户、物品的平均评分。更加可靠的方法是用内容信息去填补缺失的评分[16, 25, 41, 50]。例如,缺失评分可以由一个叫 filterbots 的自动组件提供[25, 41],这个组件扮演系统中普通用户角色,同时会根据物品具体内容的特点进行评分。这些缺失的评分也可以通过基于内容方法预测的评分来替代[50],例如,4.1.2.1节所描述的方法。最后,内容相似性也可以在系统预测中用来代替或补充评分关联相似性,找到预测时用到的最近邻[7, 46, 59, 72]。

上述提到方法同样具有一些缺陷。例如,用默认数据去填补缺失数据会导致推荐的一些偏差。同样正如4.1.2.1节所提到的,物品内容也可能不适于计算评分或者相似性。本节将描述两种解决受限覆盖或者稀疏问题的方法:降维方法和基于图(graph-based)方法。

### 4.4.1 降维方法

降维方法[4, 7, 23, 42, 67, 74, 75]解决受限覆盖和数据稀疏性问题,是通过将用户或者物品映射到隐变量空间以获取它们之间最突出的特征。因为用户或者物品之间的对比是在高级特征的密集子空间内,而不是之前的评分空间,这样更多有意义的关联就能够发现。特别地,即使用户所评的物品不相同,用户之间的关联也能够被发现。并且这类方法则会对稀疏数据不敏感[4, 7, 67]。主要有两种降维方法可以用来改善推荐系统:1)对基于用户评分矩阵进行分解;2)对稀疏的相似性矩阵进行分解。

#### 4.4.1.1 对评分矩阵进行分解

隐式语义索引(Latent semantic indexing, LSI)是对物品推荐的降维的一种流行方法[15]。在这种方法中,一个 $|U| \times |I|$ 用户-物品评分矩阵 $R$ (秩为 $n$ )可以近似表示为矩阵 $\hat{R} = PQ^T$ (秩为 $k < n$ ),其中 $P$ 是一个 $|U| \times k$ 用户因子的矩阵, $Q$ 是一个 $|I| \times k$ 物品因子的矩阵。直观地,矩阵 $P$ 的第 $u$ 行( $p_u \in \mathbf{R}^k$ )表示用户 $u$ 映射到 $k$ 维隐空间的坐标。同样的矩阵 $Q$ 的第 $i$ 行( $q_i \in \mathbf{R}^k$ )表示物品映射到 $k$ 维隐空间的坐标。矩阵 $P$ 和 $Q$ 通常通过最小化平方 Frobenius 范数定义的重建误差获取:

$$\text{err}(P, Q) = \|R - PQ^T\|_F^2 = \sum_{u,i} (r_{ui} - p_u q_i^T)^2$$

最小化这个误差等同于求解 $R$ 的奇异值分解(SVD)[24]:

$$R = U \Sigma V^T$$

其中矩阵 $U$ 是左奇异向量 $|U| \times n$ 矩阵,矩阵 $V$ 是右奇异向量 $|I| \times n$ 矩阵, $\Sigma$ 是奇异向量的 $n \times n$ 对角矩阵。选择前 $k$ 个最大奇异值及对应的奇异向量矩阵表示为 $\sum_k, U_k, V_k$ ,那么对应的用户因子矩阵和物品因子矩阵可以表示为: $P = U_k \sum_k^{1/2}, Q = V_k \sum_k^{1/2}$ 。

一旦计算得到矩阵 $P$ 和 $Q$ ,那么基于模型的评分预测 $r_{ui}$ 为

$$r_{ui} = p_u q_i^T$$

但是,使用SVD方法分解矩阵 $R$ 会带来一个严重问题:矩阵 $R$ 中多数的 $r_{ui}$ 的值是没有定义的,因为用户 $u$ 可能没有对物品 $i$ 评分。尽管可以像前面提到的那样给予 $r_{ui}$ 缺省值,这会导致数据的偏差。更加重要的是这样做会使得大矩阵 $R$ 稠密,并且使SVD分解失效。对于这类问题一个共同的做法就是在学习矩阵 $P$ 和 $Q$ 的时候仅仅使用一些已知的评分[4, 42, 73, 75]:

$$\text{err}(P, Q) = \sum_{r_{ui} \in \mathcal{R}} (r_{ui} - p_u q_i^T)^2 + \lambda (\|p_u\|^2 + \|q_i\|^2) \quad (4.29)$$

其中 $\lambda$ 是控制正则化程度的参数。在本书第5章可以找到有关这种推荐方法更详细的描述。

在基于近邻推荐方法中,也可以用这种方法计算在隐变量空间中用户或者物品的相似性[7]。通过解决以下问题来完成:

$$\begin{aligned} \text{err}(P, Q) &= \sum_{r_{ui} \in \mathcal{R}} (z_{ui} - p_u q_i^T)^2 \\ \text{s. t. } &\|p_u\| = 1, \forall u \in U, \|q_i\| = 1, \forall i \in I \end{aligned} \quad (4.30)$$

其中 $z_{ui}$ 是 $r_{ui}$ 对进行了均值中心化后使其范围在 $[-1, 1]$ 。例如,如果 $r_{\min}$ 和 $r_{\max}$ 分别为原始评分范围的最小值和最大值,那么

$$z_{ui} = \frac{r_{ui} - \bar{r}_u}{r_{\max} - r_{\min}}$$

问题转化成,对每个用户 $u$ 和物品 $i$ 而言,找到其在 $k$ 维单元球面上的坐标,如果它们的坐标在球面上距离很近,用户 $u$ 就会为物品 $i$ 评高分。如果两个用户 $u$ 和 $v$ 在球面上越接近,那么他们会给予同一物品相似的评分。因此,用户之间的相似度可以这样计算:

$$w_{uv} = p_u p_v^T$$

同样,两个物品 $i$ 和 $j$ 之间的相似也可以这样计算获得

$$w_{ij} = q_i q_j^T$$

#### 4.4.1.2 对相似性矩阵进行分解

第二种降维方法的原则和前面提到方法一样：都是将矩阵分解为因子来表示用户或者物品所映射到的隐空间。不过，这里分解的是一个稀疏的相似性矩阵而不是评分矩阵。设  $W$  为代表用户或物品相似度的秩为  $n$  的对称矩阵。为了简化这种表示，我们会假设和前面例子一样，通过最小化下面的目标函数，用更低秩  $k(k < n)$  的矩阵  $\hat{W} = PP^T$  来近似矩阵  $W$ ：

$$\text{err}(P) = \|R - PP^T\|_F^2 = \sum_{u,v} (w_{uv} - p_u p_v^T)^2$$

矩阵  $\hat{W}$  可以视为矩阵  $W$  的压缩版本，相比  $W$  不那么稀疏。和前面方法一样，计算因子矩阵  $P$  可以等同于计算矩阵  $W$  的特征值分解：

$$W = V\Lambda V^T$$

其中  $\Lambda$  是包含了  $W$  的  $|u|$  个特征值的对角矩阵， $V$  是包含了对应特征向量的  $|u| \times |u|$  的正交矩阵。设  $V_k$  为  $W$  的  $k$  个主特征向量(标准化特征向量)组成的矩阵，对应着  $k$  维的隐变量子空间的轴。矩阵  $P = V_k \Lambda_k^{1/2}$  的第  $u$  行表示用户  $u$  在这个子空间中的坐标  $p_u \in \mathbb{R}^k$ 。因此，用户在隐式子空间中的相似性可以通过以下矩阵获得

$$W' = PP^T = V_k \Lambda_k V_k^T \quad (4.31)$$

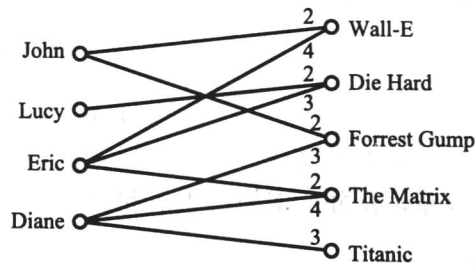
这种方法可以用于 Eigentaste 系统中推荐笑话任务[23]。在 Eigentaste 系统中，将包含了物品对之间皮尔逊关联相似度的矩阵  $W$  进行分解，可以得到由  $W$  的两个主特征向量定义的隐式子空间。令  $V_2$  表示包含这些特征向量的矩阵。将评分矩阵  $R$  的第  $u$  行  $r_u$  所代表的用户  $u$  投影到  $V_2$  所定义的平面上，得

$$r'_u = r_u V_2$$

在离线阶段中，可以通过递归细分的技术将系统中的用户在平面中进行聚类。然后，用户  $u$  对物品  $i$  的评分可以通过计算和用户  $u$  同一个聚类中的所有用户对物品  $i$  的平均评分来获得。

#### 4.4.2 基于图方法

基于图方法是将数据用图的形式表示出来，在图中用户、物品或者两者都可以用点的形式表示，边表示用户和物品之间的交互或者相似性。例如，在图 4.4 中数据可以建模成为一个二分图，其中两个节点子集分别表示用户和物品，连接用户  $u$  到物品  $i$  的边表示用户  $u$  对物品  $i$  进行了评分。可以给予边权重表示对应的评分情况。在其他一些模型里，节点集合可以只是表示成用户或者物品，连接两点的边则表示对应这些节点评分有充分的关联，而边上面的权重则表示这些关联的值。



在这些模型中，传统的方法为用户  $u$  对物品  $i$  图 4.4 用一个二分图表示图 4.1 中的评分(仅仅列出集合  $\{2, 3, 4\}$  中的评分)进行预测，仅仅会使用到这些有直接连接的用户  $u$  或者物品  $i$  的节点。基于图方法，则从另一方面出发，通过信息传递考虑非直接连接的节点间的影响。边的权重越大，越多的信息可以从此边通过。同样地，如果在图中距离较远，那么一个节点到另一个节点影响将会更小。这两个性质就是熟知的传播(propagation)和衰减(attenuation)[26, 34]，常常出现在基于

图的相似性计算当中。

基于图方法的关联传递,可以通过两种方式来推荐物品。在第一种方法中,图中用户  $u$  对物品  $i$  的相近距离可以直接用于评估用户  $u$  对物品  $i$  的评分[19, 26, 34]。通过这种思想,在系统中为用户推荐物品,可以通过找到用户  $u$  在图中最近的物品。另一方面,第二种方法将用户间或者物品间的相近距离看成他们之间的相似度权重  $w_{uv}$  或者  $w_{ij}$ , 然后使用基于近邻的推荐方法[19, 48]。

#### 4.4.2.1 基于路径的相似度

在基于路径的相似度计算中,图中两个节点间的距离可以通过计算用连接两个节点的路径数目和这些路径长度所构造的函数来获得。

##### 最短路径

文献[2]描述了一种在图模型中通过用户节点最短距离来计算他们之间相似性的推荐方法。在这种方法中,数据可以建模成为一个以用户为节点的有向图,他们之间的边是基于 horting 和可预测性所决定的。Horting 是指当两个用户对相同物品评分时才会满足条件的一种非对称关系。用公式化表示,如果用户  $u$  horting 用户  $v$ , 那么条件  $|I_{uv}| > \alpha$  或者条件  $|I_{uv}|/|I_u| > \beta$  之一需要满足,其中  $\alpha$ 、 $\beta$  是预定义的阈值。另一方面,可预见性是一个更强的属性,在将用户  $u$  和用户  $v$  的评分刻度间的差异进行转换映射后,额外要求用户  $u$  的评分和用户  $v$  的都相似。因此,用户  $v$  预测用户  $u$ , 需要假定用户  $u$  horting 用户  $v$ , 且存在一个线性转换:  $l: S \rightarrow S$ , 这样才满足以下式子:

$$\frac{1}{|I_{uv}|} \sum_{i \in I_{uv}} |r_{ui} - l(r_{vi})| \leq \gamma$$

其中  $\gamma$  是预定义的阈值。

可预测性的关系在图中表示成有向边,如边从  $u$  指向  $v$  表示  $v$  预测  $u$ 。相应地,连接用户  $u$  和用户  $v$  的有向路径表示,在经历一系列转换之后,用户  $v$  对用户  $u$  评分的传递可预测性。通过这种思想,采用从用户  $u$  到其他对新物品  $i$  评分用户的最短有向路径,就能预测出用户  $u$  对新物品  $i$  的评分。令  $P = \{u, v_1, v_2, \dots, v_m\}$  表示路径集合,其中  $v_m \in U_i$ 。根据这条路径上线性映射的组合,可以将用户  $v_m$  对物品  $i$  的评分转换到用户  $u$  的评分刻度上:

$$\hat{r}_{ui}^{(P)} = (l_m \circ \dots \circ l_2 \circ l_1)(r_{vi})$$

最后,预测评分  $r_{ui}$  可以通过计算所有最短路径  $P$  中的  $\hat{r}_{ui}^{(P)}$  的平均值。

##### 路径数量

在二分图中用户与物品之间的路径数量可以用来评估他们之间的连通性[34]。令  $R$  表示一个大小为  $|U| \times |I|$  的评分矩阵,其中如果用户  $u$  对物品  $i$  进行了评分  $r_{ui}$  等于 1, 否则等于 0。二分图的邻接矩阵可以通过  $R$  来定义:

$$A = \begin{pmatrix} 0 & R^T \\ R & 0 \end{pmatrix}$$

在这种方法中,用户  $u$  与物品  $i$  之间的关联度可以定义为用户  $u$  到用户  $v$  的所有不同路径的权重之和(允许每个节点在图中出现超过一次),其中长度不能超过给出的最大长度  $K$ 。由于图是二分图,所以  $K$  的值应该是一个奇数。为了减弱距离较长路径的贡献度,可以给出一个关于长度  $K$  的权重  $\alpha^k$ , 其中  $\alpha \in [0, 1]$ 。由于  $A^k$  给出了节点对之间长度为  $k$  的路径数量,那么用户—物品的关联矩阵  $S_k$  可以这样表示:



$$S_K = \sum_{k=1}^K \alpha^k A^k = (I - \alpha A)^{-1} (\alpha A - \alpha^K A^K) \quad (4.32)$$

这种在图中计算节点之间距离的方法就是为人熟知的 Katz 计算方法[38]。这种方法与冯诺依曼扩散(Von Neumann Diffusion)核方法[20, 40, 43]:

$$K_{\text{VND}} = \sum_{k=0}^{\infty} \alpha^k A^k = (I - \alpha A)^{-1} \quad (4.33)$$

及指数扩散(Exponential Diffusion)核方法:

$$K_{\text{ED}} = \sum_{k=0}^{\infty} \frac{1}{k!} \alpha^k A^k = \exp(\alpha A) \quad (4.34)$$

紧密关联, 其中  $A^0 = I$ 。

在推荐系统中包含了大量的用户和物品, 计算他们的关联值需要大量的计算资源。为了克服这种限制, 文献[34]中使用了扩散激活(spreading activation)技术[14]。本质上, 这类技术首先激活一部分节点子集作为开始节点, 然后迭代地激活那些和已经激活节点直接连接的节点, 直到达到收敛的标准。

#### 4.4.2.2 随机游走相似性

在基于图方法里的传递关联也可以通过概率框架来定义。在这个框架里, 用户或者物品之间的相似性可以估算为到达这些点的一次随机漫步概率。公式化地描述为一个一阶的马尔可夫随机过程, 其定义会用到一组  $n$  个状态, 以及一个  $n \times n$  的转移概率矩阵  $P$ , 其中在任意时段  $t$  从状态  $i$  跳转到状态  $j$  的概率是

$$p_{ij} = \Pr(s(t+1) = j \mid s(t) = i)$$

定义向量  $\pi(t)$  为包含了步骤  $t$  的状态概率分布, 因此  $\pi_i(t) = \Pr(s(t) = i)$ 。马尔可夫链过程可以这样表示:

$$\pi(t+1) = P^T \pi(t)$$

因此, 如果  $P$  的行是随机的(即对于所有  $i$ ,  $\sum_j p_{ij} = 1$ ), 这个过程收敛于一个稳定的分布向量  $\pi(\infty)$ , 这向量对应于  $P^T$  中特征值为 1 的特征向量。这个过程通常可以用每个节点对应一个状态的有权图来描述, 其中一个节点到另一个节点的跳转概率则是由连接两点的边的权重给出的。

#### ItemRank 算法

ItemRank[26]方法是基于 PageRank 算法[11](对 Web 网页进行排行的算法)的一种推荐方法。这种方法基于用户在图中随机游走访问到物品  $i$  的概率, 对用户  $u$  对新物品  $i$  的喜爱程度进行排序, 其中图的节点表示评分物品, 有相同用户评分的节点间用边相连。边的权重通过一个  $|\mathcal{I}| \times |\mathcal{I}|$  的转换概率矩阵给出, 其中  $p_{ij} = |u_{ij}| / |u_i|$  表示当用户已经对物品  $i$  评分后还会对物品  $j$  评分的预估条件概率。

在 PageRank 算法中, 对于任意一步  $t$  的随机游走, 既可以通过  $P$  以固定的概率  $\alpha$  跳转到邻近节点, 也可以  $(1-\alpha)$  的概率“瞬移”到任意节点。令  $r_u$  表示评分矩阵  $R$  的第  $u$  行, 用户  $u$  瞬移到其他节点的概率分布函数可以通过向量  $d_u = r_u / \|r_u\|$  给出。通过上面的定义, 用户  $u$  在第  $t+1$  步状态概率分布向量可以这样表示:

$$\pi_u(t+1) = \alpha P^T \pi_u(t) + (1-\alpha) d_u \quad (4.35)$$

在实际应用中, 通常这样计算  $\pi_u(\infty)$ : 首先使用均匀分布初始化, 如  $\pi_u(0) = \frac{1}{n} \mathbf{1}_n$ , 然后使用式(4.35)更新  $\pi_u$ , 直到其收敛。一旦计算获得  $\pi_u(\infty)$ , 系统就会向用户  $u$  推荐

$\pi_u$  值最大的物品  $i$ 。

### 平均首次通过/往返次数

针对推荐问题还有其他基于随机游走的距离测量方法。其中包括平均首次通过次数和平均往返次数[19, 20]。平均首次通过次数(average first-passage time) $m(j|i)$ [56]是指当开始节点  $i \neq j$  时, 第一次到达节点  $j$  所需要平均随机游走步数。令  $P$  表示为一个  $n \times n$  的概率转换矩阵, 那么  $m(j|i)$  可以这样计算获得

$$m(j|i) = \begin{cases} 0, & \text{若 } i = j \\ 1 + \sum_{k=1}^n p_{ik} m(j|k), & \text{其他} \end{cases}$$

平均首次通过次数方法的一个问题是它不对称, 一个相关的解决方法是平均往返次数(average commute time),  $n(i, j) = m(j|i) + m(i|j)$ [22], 表示从开始节点  $i \neq j$  第一次到达节点  $j$  并且返回到  $i$  的平均随机游走步数。这种计算方法有几个有意思的性质, 也就是说, 这是某种欧几里得空间中的真实距离[22], 这个性质和电力网络中电阻性质以及 Laplacian 矩阵图中的伪逆性质接近[19]。

在文献[19]中, 平均往返次数用来计算推荐系统中代表了用户和物品间相互作用的二分图里的节点之间的距离。对于每一个用户  $u$  到任意一个物品  $i \in I_u$  的有向边的权重可以简单表示为  $1/|I_u|$ 。同样地, 对于每一个物品  $i$  到任意一个物品  $u \in U_i$  的有向边, 其权重为  $1/|U_i|$ 。平均往返次数可以用在两个不同地方: 1) 依据  $n(u, i)$  的极小值为用户  $u$  推荐物品  $i$ ; 2) 根据往返次数的距离, 寻找用户  $u$  的近邻用户, 然后为用户  $u$  推荐这些用户喜爱的物品。

## 4.5 总结

基于近邻推荐方法是物品推荐任务中最早使用的方法之一, 现在依然有着相当广泛的应用。尽管这种方法可以简单地描述和实现, 但它有着几个重要的优势: 它可以解释一个推荐任务为什么使用列表中的这些近邻, 其计算和空间的效率使其可扩展到大的推荐系统中, 线上加入新的用户或者物品能够保持稳定性。这种方法另外一个重要优势是使得推荐拥有新颖性, 使得用户可以发现没有期望得到但却是感兴趣的物品。

实现基于近邻推荐方法的时候, 需要做几个重要决策。如选择基于用户或者选择基于物品推荐方法会很大程度上影响推荐系统的准确性和效率。在传统的商业推荐系统中, 用户的数量要远远大于物品的有效数量, 选择基于物品方法会更加适合, 因为它们可以提供更加准确的推荐, 同时计算效率更高且很少需要频繁的更新。另一方面, 基于用户推荐方法通常可以给出更加独到的推荐, 这种方法会使得用户更加满意。另外, 基于近邻方法的不同组成部分, 如评分标准化、相似性权重的计算、近邻的选择, 都会对推荐系统的质量产生重要的影响。对于每一个组成部分, 都有几种不同的方法可供选择, 尽管每种方法的优点中在文献中描述了, 但是需要注意的是, 在不同的系统中, 最适合的方法也是不相同的, 因此, 从实际的系统和应用的需要出发才能对这些算法做出最有效的评价。

最后, 当基于近邻方法的效果受到受限覆盖和数据稀疏性问题影响的时候, 可以尝试基于降维和基于图的方法。降维方法可以提供用户和物品的一个压缩表示, 这样就可以获取他们最重要的特征。这类方法的一个重要优点在于, 即使用户对不同物品进行了评分或

物品被不同的用户所评分, 仍可以获取用户间或者物品间的有效关联。另一方面, 基于图模型方法是挖掘数据之间的传递关系。这种技术可以通过计算用户之间或者物品之间非直接的关联来有效避免数据稀疏和覆盖受限问题。同时, 与降维方法不同的是, 基于图方法可以保留数据之间的一些“局部”关联, 从而可以提供新颖的推荐。

## 参考文献

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
2. Aggarwal, C.C., Wolf, J.L., Wu, K.L., Yu, P.S.: Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In: *KDD '99: Proc. of the 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 201–212. ACM, New York, NY, USA (1999)
3. Balabanović, M., Shoham, Y.: Fab: Content-based, collaborative recommendation. *Communications of the ACM* 40(3), 66–72 (1997)
4. Bell, R., Koren, Y., Volinsky, C.: Modeling relationships at multiple scales to improve accuracy of large recommender systems. In: *KDD '07: Proc. of the 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 95–104. ACM, New York, NY, USA (2007)
5. Bell, R.M., Koren, Y.: Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In: *ICDM '07: Proc. of the 2007 Seventh IEEE Int. Conf. on Data Mining*, pp. 43–52. IEEE Computer Society, Washington, DC, USA (2007)
6. Billsus, D., Brunk, C.A., Evans, C., Gladish, B., Pazzani, M.: Adaptive interfaces for ubiquitous web access. *Communications of the ACM* 45(5), 34–38 (2002)
7. Billsus, D., Pazzani, M.J.: Learning collaborative information filters. In: *ICML '98: Proc. of the 15th Int. Conf. on Machine Learning*, pp. 46–54. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
8. Billsus, D., Pazzani, M.J.: User modeling for adaptive news access. *User Modeling and User-Adapted Interaction* 10(2-3), 147–180 (2000)
9. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *Journal of Machine Learning Research* 3, 993–1022 (2003)
10. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: *Proc. of the 14th Annual Conf. on Uncertainty in Artificial Intelligence*, pp. 43–52. Morgan Kaufmann (1998)
11. Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30(1-7), 107–117 (1998)
12. Buckley, C., Salton, G.: Optimization of relevance feedback weights. In: *SIGIR '95: Proc. of the 18th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 351–357. ACM, New York, NY, USA (1995)
13. Cohen, W.W., Schapire, R.E., Singer, Y.: Learning to order things. In: *NIPS '97: Proc. of the 1997 Conf. on Advances in Neural Information Processing Systems*, pp. 451–457. MIT Press, Cambridge, MA, USA (1998)
14. Crestani, F., Lee, P.L.: Searching the Web by constrained spreading activation. *Information Processing and Management* 36(4), 585–605 (2000)
15. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. *Journal of the American Society for Information Science* 41, 391–407 (1990)
16. Degemmis, M., Lops, P., Semeraro, G.: A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation. *User Modeling and User-Adapted Interaction* 17(3), 217–255 (2007)
17. Delgado, J., Ishii, N.: Memory-based weighted majority prediction for recommender systems. In: *Proc. of the ACM SIGIR'99 Workshop on Recommender Systems* (1999)
18. Deshpande, M., Karypis, G.: Item-based top-N recommendation algorithms. *ACM Transaction on Information Systems* 22(1), 143–177 (2004)
19. Fouss, F., Renders, J.M., Pirotte, A., Saeuens, M.: Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering* 19(3), 355–369 (2007)
20. Fouss, F., Yen, L., Pirotte, A., Saeuens, M.: An experimental investigation of graph kernels on a collaborative recommendation task. In: *ICDM '06: Proc. of the 6th Int. Conf. on Data Mining*, pp. 863–868. IEEE Computer Society, Washington, DC, USA (2006)

21. Freund, Y., Iyer, R.D., Schapire, R.E., Singer, Y.: An efficient boosting algorithm for combining preferences. In: ICML '98: Proc. of the 15th Int. Conf. on Machine Learning, pp. 170–178. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
22. Gobel, F., Jagers, A.: Random walks on graphs. *Stochastic Processes and Their Applications* **2**, 311–336 (1974)
23. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval* **4**(2), 133–151 (2001)
24. Golub, G.H., Van Loan, C.F.: *Matrix computations* (3rd ed.). Johns Hopkins University Press (1996)
25. Good, N., Schafer, J.B., Konstan, J.A., Borchers, A., Sarwar, B., Herlocker, J., Riedl, J.: Combining collaborative filtering with personal agents for better recommendations. In: AAAI '99/IAAI '99: Proc. of the 16th National Conf. on Artificial Intelligence, pp. 439–446. American Association for Artificial Intelligence, Menlo Park, CA, USA (1999)
26. Gori, M., Pucci, A.: Itemrank: a random-walk based scoring algorithm for recommender engines. In: Proc. of the 2007 IJCAI Conf., pp. 2766–2771 (2007)
27. Grcar, M., Fortuna, B., Mladenic, D., Grobelnik, M.: k-NN versus SVM in the collaborative filtering framework. *Data Science and Classification* pp. 251–260 (2006). URL <http://db.cs.ualberta.ca/webkdd05/proc/paper25-mladenic.pdf>
28. Herlocker, J., Konstan, J.A., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Inf. Retr.* **5**(4), 287–310 (2002)
29. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR '99: Proc. of the 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 230–237. ACM, New York, NY, USA (1999)
30. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* **22**(1), 5–53 (2004)
31. Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: CHI '95: Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, pp. 194–201. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995)
32. Hofmann, T.: Collaborative filtering via Gaussian probabilistic latent semantic analysis. In: SIGIR '03: Proc. of the 26th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 259–266. ACM, New York, NY, USA (2003)
33. Howe, A.E., Forbes, R.D.: Re-considering neighborhood-based collaborative filtering parameters in the context of new data. In: CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management, pp. 1481–1482. ACM, New York, NY, USA (2008)
34. Huang, Z., Chen, H., Zeng, D.: Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems* **22**(1), 116–142 (2004)
35. Jin, R., Chai, J.Y., Si, L.: An automatic weighting scheme for collaborative filtering. In: SIGIR '04: Proc. of the 27th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 337–344. ACM, New York, NY, USA (2004)
36. Jin, R., Si, L., Zhai, C.: Preference-based graphic models for collaborative filtering. In: Proc. of the 19th Annual Conf. on Uncertainty in Artificial Intelligence (UAI-03), pp. 329–33. Morgan Kaufmann, San Francisco, CA (2003)
37. Jin, R., Si, L., Zhai, C., Callan, J.: Collaborative filtering with decoupled models for preferences and ratings. In: CIKM '03: Proc. of the 12th Int. Conf. on Information and Knowledge Management, pp. 309–316. ACM, New York, NY, USA (2003)
38. Katz, L.: A new status index derived from sociometric analysis. *Psychometrika* **18**(1), 39–43 (1953)
39. Kendall, M., Gibbons, J.D.: *Rank Correlation Methods*, 5 edn. Charles Griffin (1990)
40. Kondor, R.I., Lafferty, J.D.: Diffusion kernels on graphs and other discrete input spaces. In: ICML '02: Proc. of the Nineteenth Int. Conf. on Machine Learning, pp. 315–322. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2002)
41. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to usenet news. *Communications of the ACM* **40**(3), 77–87 (1997)
42. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD'08: Proceeding of the 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, pp. 426–434. ACM, New York, NY, USA (2008)
43. Kunegis, J., Lommatzsch, A., Bauckhage, C.: Alternative similarity functions for graph kernels. In: Proc. of the Int. Conf. on Pattern Recognition (2008)

44. Lang, K.: News Weeder: Learning to filter netnews. In: Proc. of the 12th Int. Conf. on Machine Learning, pp. 331–339. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA (1995)
45. Last.fm: Music recommendation service (2009). <http://www.last.fm>
46. Li, J., Zaiane, O.R.: Combining usage, content, and structure data to improve Web site recommendation. In: Proc. of the 5th Int. Conf. on Electronic Commerce and Web Technologies (EC-Web) (2004)
47. Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* 7(1), 76–80 (2003)
48. Luo, H., Niu, C., Shen, R., Ullrich, C.: A collaborative filtering framework based on both local user similarity and global user similarity. *Machine Learning* 72(3), 231–245 (2008)
49. Ma, H., King, I., Lyu, M.R.: Effective missing data prediction for collaborative filtering. In: SIGIR '07: Proc. of the 30th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 39–46. ACM, New York, NY, USA (2007)
50. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations. In: 18th National Conf. on Artificial intelligence, pp. 187–192. American Association for Artificial Intelligence, Menlo Park, CA, USA (2002)
51. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J.: MovieLens unplugged: experiences with an occasionally connected recommender system. In: IUI '03: Proc. of the 8th Int. Conf. on Intelligent User Interfaces, pp. 263–266. ACM, New York, NY, USA (2003)
52. Mobasher, B., Dai, H., Luo, T., Nakagawa, M.: Discovery and evaluation of aggregate usage profiles for Web personalization. *Data Mining and Knowledge Discovery* 6(1), 61–82 (2002)
53. Mooney, R.J.: Content-based book recommending using learning for text categorization. In: Proc. of the Fifth ACM Conf. on Digital Libraries, pp. 195–204. ACM Press (2000)
54. Nakamura, A., Abe, N.: Collaborative filtering using weighted majority prediction algorithms. In: ICML '98: Proc. of the 15th Int. Conf. on Machine Learning, pp. 395–403. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
55. Netflix: Online movie rental service (2009). <http://www.netflix.com>
56. Norris, J.R.: Markov Chains, 1 edn. Cambridge University Press, Cambridge (1999)
57. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In: Proceedings of the KDD Cup and Workshop (2007)
58. Pazzani, M., Billsus, D.: Learning and revising user profiles: The identification of interesting Web sites. *Machine Learning* 27(3), 313–331 (1997)
59. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review* 13(5-6), 393–408 (1999)
60. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: An open architecture for collaborative filtering of netnews. In: CSCW '94: Proc. of the 1994 ACM Conf. on Computer Supported Cooperative Work, pp. 175–186. ACM, New York, NY, USA (1994)
61. Rich, E.: User modeling via stereotypes. *Cognitive Science* 3(4), 329–354 (1979)
62. Rissanen, J.: Modeling by shortest data description. *Automatica* 14, 465–471 (1978)
63. Rocchio, J.: Relevance Feedback in Information Retrieval. Prentice Hall, Englewood, Cliffs, New Jersey (1971)
64. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted Boltzmann machines for collaborative filtering. In: ICML '07: Proceedings of the 24th international conference on Machine learning, pp. 791–798. ACM, New York, NY, USA (2007)
65. Salton, G. (ed.): Automatic text processing. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1988)
66. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW '01: Proc. of the 10th Int. Conf. on World Wide Web, pp. 285–295. ACM, New York, NY, USA (2001)
67. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.: Application of dimensionality reduction in recommender systems: A case study. In: ACM WebKDD Workshop (2000)
68. Sarwar, B.M., Konstan, J.A., Borchers, A., Herlocker, J., Miller, B., Riedl, J.: Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In: CSCW '98: Proc. of the 1998 ACM Conf. on Computer Supported Cooperative Work, pp. 345–354. ACM, New York, NY, USA (1998)
69. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: SIGIR '02: Proc. of the 25th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 253–260. ACM, New York, NY, USA (2002)
70. Shardanand, U., Maes, P.: Social information filtering: Algorithms for automating “word of mouth”. In: CHI '95: Proc. of the SIGCHI Conf. on Human factors in Computing Systems,



- pp. 210–217. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995)
71. Sheth, B., Maes, P.: Evolving agents for personalized information filtering. In: Proc. of the 9th Conf. on Artificial Intelligence for Applications, pp. 345–352 (1993)
  72. Soboroff, I.M., Nicholas, C.K.: Combining content and collaboration in text filtering. In: Proc. of the IJCAI'99 Workshop on Machine Learning for Information Filtering, pp. 86–91 (1999)
  73. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Major components of the gravity recommendation system. SIGKDD Exploration Newsletter 9(2), 80–83 (2007)
  74. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Investigation of various matrix factorization methods for large recommender systems. In: Proc. of the 2nd KDD Workshop on Large Scale Recommender Systems and the Netflix Prize Competition (2008)
  75. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Scalable collaborative filtering approaches for large recommender systems. Journal of Machine Learning Research (Special Topic on Mining and Learning with Graphs and Relations) 10, 623–656 (2009)
  76. Terveen, L., Hill, W., Amento, B., McDonald, D., Creter, J.: PHOAKS: a system for sharing recommendations. Communications of the ACM 40(3), 59–62 (1997)
  77. Zhang, Y., Callan, J., Minka, T.: Novelty and redundancy detection in adaptive filtering. In: SIGIR '02: Proc. of the 25th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 81–88. ACM, New York, NY, USA (2002)
  78. Zitnick, C.L., Kanade, T.: Maximum entropy for collaborative filtering. In: AUAI '04: Proc. of the 20th Conf. on Uncertainty in Artificial Intelligence, pp. 636–643. AUAI Press, Arlington, Virginia, United States (2004)

## 协同过滤算法的高级课题

Yehuda Koren 和 Robert Bell

**摘要** 推荐系统的协同过滤算法已经得到了人们的普遍关注并取得了很大的进展。协同过滤算法在近期结束的 Netflix 推荐系统比赛中起到的核心作用无疑又增加了它的知名度。本章展示了协同过滤算法领域的新进展。矩阵因子分解技术已经成为实现协同过滤算法的首选，本章描述了该技术及其最新改进。此外，本章也描述了协同过滤领域曾经处于统治地位的基于邻域的算法的一些扩展，这些扩展提高了这些传统算法的准确性，使得它们更具竞争力。本章阐述了如何利用时态模型和隐反馈来提升模型的准确性。另外，本章对参赛者为解决 Netflix Prize 推荐系统比赛的挑战而提出的一些主要的推荐算法给出了详细的阐述。

### 5.1 简介

协同过滤(CF)方法基于用户对商品的评分或其他行为(如购买)模式来为用户提供个性化的推荐，而不需要了解用户或者商品的大量信息。尽管目前已经成型的方法在很多应用上都工作得很好，我们依然提出了一些最近提出的对该领域已有方法的拓展，旨在为正在寻找最有可能成功的推荐的分析员提供一些帮助。

Netflix Prize 比赛于 2006 年开始举行，该比赛对协同过滤领域的发展起到了重要的推动作用。科研界第一次获得了大规模的工业界数据(数以亿计的电影评分)，这吸引了数以千计的科学家、学生、工程师以及推荐系统的狂热者进入协同过滤研究领域。比赛的性质使得该领域迅速发展，新技术层出不穷，而且每一次技术的变革都旨在提高预测准确度。由于所有的方法都在相同的数据上经同样的严格标准来评判，演进而来的模型功能更强大、更高效。

推荐系统的性能依赖于输入的不同类型。最高效的输入是用户高质量的显式反馈，也就是用户直接说出他们对哪些产品感兴趣。例如，Netflix 通过让用户单击“大拇指”或“下拇指”来搜集电影的星级评价。同样，TiVo 用户也用这一方式来表明他们对电视节目的喜好。

由于显式反馈并不经常可用，一些推荐系统从比较丰富的隐式反馈中来推断用户的爱

---

Yehuda Koren, Yahoo! Research, e-mail: yehuda@yahoo-inc.com

Robert Bell, AT&T Labs- Research e-mail: rbell@research.att.com

This article includes copyrighted materials, which were reproduced with permission of ACM and IEEE. The original articles are;

R. Bell and Y. Koren, "Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights", *IEEE International Conference on Data Mining (ICDM'07)*, pp. 43-52, © 2007 IEEE. Reprinted by permission.

Y. Koren, "Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model", *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, © 2008 ACM, Inc.

Reprinted by permission. <http://doi.acm.org/10.1145/1401890.1401944>

翻译: 王二朋 审核: 吴亮, 吴涛, 郑州大学-吴宾

好,也就是通过观察用户的行为来间接得到用户的喜好[22]。隐式反馈的类型包括用户的购买记录、浏览历史记录、搜索模式,甚至是鼠标的移动。例如,如果一个人购买了同一个作者的许多书则可以说明这个人很喜欢该作者。本章重点关注与显式反馈相匹配的模型。尽管如此,我们也意识到隐式反馈的重要性。当用户没有提供显式反馈的时候,隐式反馈就显得弥足珍贵了。因此,我们也会展示如何在模型中把隐式反馈当作辅助信息来处理。

为了建立推荐信息,CF系统需要把两种有本质区别的实体联系起来,即物品和用户。有两种主要的方法来方便比较这两种实体,构成了CF的两种主要技术:基于邻域的方法和隐语义模型。基于邻域的方法重点关注物品之间的关系或者用户之间的关系。基于物品的方法是根据某用户对和他感兴趣的物品相似的物品的评分,来对该用户对这件物品的偏好建立模型。隐语义模型,如矩阵因子分解模型,采用的是另一种方法,即把物品与用户映射到相同的隐语义空间的方法。隐语义空间试图通过描述产品和用户两种实体在因子上的特征来解释评分,而这些因子是根据用户的反馈自动推断出来的。

产生具有更精确预测结果的方法需要深化基础和减少对任意决策的依赖。本章阐述了一系列近期提出的基本CF建模技术的改进。然而,要想追求更加精确的模型不能满足和局限于这些改进。同等重要的是对数据中所有可用的信号或者特征的识别。传统的技术解决了用户—物品评分数据中的稀疏问题。利用其他的信息可以显著提高精确性。一个典型的案例是考虑各种时间效应信息,这些时间效应信息反映了用户—物品交互的、动态的时间漂移特性。相对不那么重要的是监听用户的隐式反馈,如用户对哪些产品来评分(不管评分分数)。被评分的物品不是随机选择的,而是揭示了用户感兴趣的方面,而这些信息比评分的数值更加重要。

5.3节阐述了矩阵因子分解的一些技术,这些技术综合考虑了算法实现的便利性和相对高的预测准确度。这也使得这些技术成为了解决最大的公开数据集——Netflix数据的推荐问题的最佳算法。这一节描述了这些技术背后的理论和实用细节。除此之外,矩阵因子分解模型的优势的大部分源于它们处理数据的额外特征的能力,这些额外特征包括隐式反馈和时序信息。本节描述了如何增强矩阵分解模型来处理这些额外特征。

5.4节讨论了基于邻域的方法。这个流派中的基本方法是众所周知的,而且在很大程度上是基于启发式方法的。近期提出的一些技术提议使用更加严格的公式来克服基于邻域的方法的缺点,因此提高了预测准确度。5.5节阐述了一种更高级的方法,这个方法基于普通基于邻域的方法的原理,并使用了因式分解模型的典型技术——全局优化技术。这个方法允许放宽对邻域大小的限制,同时考虑了隐式反馈和时间效应信息。该方法预测结果的精度接近矩阵因子分解模型,同时与矩阵因子分解模型相比具有一些实用的优势。

当从最根本上考虑这些模型时,我们就会发现那些看似无关的技术之间竟然存在着惊人的联系。我们将会在第5.6节通过证明基于物品和基于用户这两种基于邻域的模型或许会在同一个模型中同时出现来详细说明这一点。而且在那个模型中,这两种模型都等价于矩阵因子分解模型。这些关联削弱了之前分类方法的相关性。比如,我们传统上把矩阵因子分解模型广义分类为“基于模型的”,而把基于邻域的模型广义分类为“基于记忆的”。

## 5.2 预备知识

现在对 $m$ 个用户(也就是消费者)和 $n$ 个物品(也就是产品)进行评分。为了区别用户和物品,我们用特殊的索引字母来具体指代二者:用 $u$ 、 $v$ 代表用户, $i$ 、 $j$ 、 $l$ 代表物品。评

分  $r_{ui}$  代表用户  $u$  对物品  $i$  的偏好程度, 值越大代表用户对物品越感兴趣。例如, 评分值可以为取值范围为 1(表示不感兴趣)到 5(表示很感兴趣)的整数。我们用  $\hat{r}_{ui}$  来表示预测的偏好程度, 用以和真实值  $r_{ui}$  区别。

数值  $t_{ui}$  表示评分  $r_{ui}$  的发生时间。该时间的单位随着当前应用的情况而变化。例如, 若以天为单位,  $t_{ui}$  记录了从早期某一时间点到现在的天数。通常绝大部分的评分是未知的。例如, 在 Netflix 数据中 99% 的可能评分是缺失的, 因为用户一般只会给予一小部分电影评分。评分值  $r_{ui}$  已知的  $(u, i)$  对存放在集合  $K = \{(u, v) : r_{uv} \text{ 已知}\}$ 。每一个用户  $u$  与表示为  $R(u)$  的一个物品集合相关联, 该集合包含了用户  $u$  给予评分的所有物品列表。与之相似的是,  $R(i)$  定义了评分了物品  $i$  的所有用户的列表。有时, 定义集合  $N(u)$  为用户  $u$  提供了隐式偏好信息的物品(如用户租用过、购买过或浏览过的物品)列表。

由评分数据建立的模型通过拟合已经观测到的评分记录来进行学习。然而, 我们的目标是以一种能预测未来未知的评分的方式来一般化这些已经观测到的结果。因此, 我们应当小心谨慎预防过拟合观测数据。我们通过正则化学习参数来避免过拟合现象, 即对这些参数的复杂度增加惩罚系数。正则化由  $\lambda_1$ 、 $\lambda_2$  等常量参数来控制。这些常量通过交叉验证来决定。随着它们的增长, 正则化任务变得越来越繁重。

### 5.2.1 基准预测

CF 模型试图捕捉用户和物品之间的交互作用, 正是这些交互作用产生了不同的评分值。然而, 大部分观察到的评分值要么与用户相关, 要么与物品相关, 而与用户和物品之间的交互作用无关。比如, 典型的 CF 数据展示了用户和物品中存在的偏置, 也就是说, 数据中有某些用户评分比其他用户高和某些物品得到的评分比其他物品高的明显倾向。

我们将把这些与用户—物品交互作用无关的因子(这些因子也叫偏置)封装到基准预测中。由于这些基准预测值在观察到的评分中占很大比例, 所以对它们进行准确的建模就显得至关重要了。这样的建模方法把真正代表用户—物品之间交互作用的那部分数据隔离开来, 而把这部分数据放到更合适的用户偏好模型中。

设  $\mu$  为总体平均评分。未知评分  $r_{ui}$  的基准预测  $b_{ui}$  综合考虑了用户和物品两个因子:

$$b_{ui} = \mu + b_u + b_i \quad (5.1)$$

参数  $b_u$  和  $b_i$  分别表示用户  $u$  和物品  $i$  的与评分平均值的偏差。例如, 假设我们想建立一个用户 Joe 对 Titanic 电影的评分的基准预测。假设所有电影的平均评分  $\mu$  为 3.7 星。另外, Titanic 比一般的电影要好, 因此其评分要比平均评分高 0.5 星。另外, Joe 是一个爱挑剔的用户, 他的评分一般要比平均评分低 0.3 星。因此, Joe 对 Titanic 评分的基准预测为  $3.7 - 0.3 + 0.5 = 3.9$ 。我们可以通过解决最小二乘法问题来估计  $b_u$  和  $b_i$  的值, 如下面公式所示:

$$\min_{b_u, b_i} \sum_{(u,i) \in K} (r_{ui} - \mu - b_u - b_i)^2 + \lambda_1 \left( \sum_u b_u^2 + \sum_i b_i^2 \right)$$

在这个公式中, 第一项  $\sum_{(u,i) \in K} (r_{ui} - \mu + b_u + b_i)^2$  用来寻找与已知评分数据拟合得最好的  $b_u$  和  $b_i$ 。正则化项  $\lambda_1 \left( \sum_u b_u^2 + \sum_i b_i^2 \right)$  通过对参数的复杂性增加惩罚因子来避免过拟合现象。这个最小二乘问题可以通过在 5.3.1 节描述的随机梯度下降算法有效地解决。

Netflix 电影评分数据的平均评分( $\mu$ )为 3.6。已知评分数据中用户偏置( $b_u$ )的均值为 0.044, 标准差为 0.41。用户偏置( $|b_u|$ )的绝对值的均值为 0.32。学习数据中物品偏置  $b_i$

的均值为  $(b_i) - 0.26$ ，标准差为 0.48。物品偏置  $(|b_i|)$  的绝对值的均值为 0.43。

一种简单但是准确度不高的估计参数的方法是把  $b_i$  的计算和  $b_u$  的计算分离开来。首先，对每一个物品  $i$ ，令

$$b_i = \frac{\sum_{u \in R(i)} (r_{ui} - \mu)}{\lambda_2 + |R(i)|}$$

对每一个用户  $u$ ，令

$$b_u = \frac{\sum_{i \in R(u)} (r_{ui} - \mu - b_i)}{\lambda_3 + |R(u)|}$$

通过使用正则化参数  $\lambda_2, \lambda_3$ ，我们可以把上述这些均值缩小至零，参数  $\lambda_2, \lambda_3$  由交叉验证决定。Netflix 数据集上通常取  $\lambda_2 = 25, \lambda_3 = 10$ 。

在 5.3.3.1 节，我们将会证明如何通过考虑数据内部的时间效应信息来提高基准预测的准确度。

## 5.2.2 Netflix 数据

为了比较本章描述的算法的相对准确性，我们在 Netflix 数据上对这些算法进行了评估，Netflix 数据是由匿名 Netflix 客户从 1999 年 11 月到 2005 年 12 月做出的具有邮戳日期的上亿条电影评分[5]。评分数据是从 1 到 5 的整数。数据由超过 480 000 用户对 17 770 部电影的评分构成。因此，平均一部电影会得到 5600 个评分，而一个用户会对 208 部电影进行评分，当然这些均值会有很大的方差。考虑到与其他方法公布的结果的兼容性，我们采用由 Netflix 设定的标准。首先，结果的质量通常由均方根误差(RMSE)来度量：

$$\sqrt{\sum_{(u,i) \in \text{TestSet}} (r_{ui} - \hat{r}_{ui})^2 / |\text{TestSet}|}$$

跟另一种度量方法(平均绝对误差)相比，这个度量更关注大误差。(参考第 8 章里面关于推荐系统评价标准的综述。)

我们在 Netflix 提供的测试集(也叫问答集)上来报告预测结果，这个测试集包含了近期 140 万条评分记录。与训练集数据相比，测试集数据包含了更多的那些不经常对电影评分的用户的评分记录，因此这些用户的评分行为也就更难预测。在某种程度上，这种情况代表了 CF 系统的真实需求，即需要从旧的评分中得到新的评分以及需要均衡地覆盖所有用户，而不只是经常做评分的那些。

Netflix 数据是 Netflix Prize 比赛的一部分，这个比赛的基准平台是 Netflix 的私有系统——Clibematch，该系统在测试集上得到的 RMSE 为 0.9514。大奖将会颁给经过三年努力把 RMSE 降低到 0.8563(在原有的基础上提高 10%)以下的团队。测试集上可得到的 RMSE 值可提升空间非常小，因此很明显要想得到这个大奖非常困难。但是，证据显示 RMSE 值的小幅度提升将会对 top-k 推荐的质量产生很大的影响[17, 19]。

## 5.2.3 隐式反馈

本章集中讨论了显式的用户反馈。但是，如果有额外的隐式反馈信息可用，这些信息可用来更好地探索用户的行为。这在解决数据的稀疏问题以及当用户的显式反馈很少时显得非常有用。我们将阐述之前讨论的模型的一些拓展，这些拓展利用了隐式反馈信息。

对于像 Netflix 这样的数据集来说，得到隐式反馈信息的最自然的方式是电影的租借



历史记录。这个历史记录告诉我们用户的爱好而不需要让他们显式地提供他们的评分。对于其他的数据集,浏览历史记录或购买历史记录可以作为隐式反馈信息。但是这些数据并不提供给我们来做实验。尽管如此,在 Netflix 数据集中还存在着一种比较隐晦的用户隐式信息。这个数据集不仅告诉我们评分值,还包括用户对哪些电影进行了评分,不论他们如何进行评分。也就是说,如果一个用户选择一部电影并打分,那么她就隐式地告知了我们她的兴趣。于是可以建立一个二元矩阵,其中 1 代表已评分,0 代表未评分。尽管这些二元数据并没有包含像其他表示隐式反馈的独立信息源那样多的信息,但是整合这些隐式的二元数据也会大大提高预测的准确度。使用这个二元数据能使预测模型受益与一个事实密不可分,这个事实就是用户的评分记录并不是随机缺失的,用户会小心谨慎地选择他们要进行评分的物品(参见 Marlin 等[21])。

### 5.3 因子分解模型

用隐语义模型来进行协同过滤的目标是揭示隐藏的特征,这些隐藏的特征能够解释观测到的评分。该模型的一些实例包括 pLSA 模型[15]、神经网络模型[24]、隐式 Dirichlet 分配模型[7],以及由用户—物品评分矩阵的因子分解推导出的模型(也叫作基于 SVD 的模型)。最近,矩阵因子分解模型由于其准确性和稳定性得到越来越多人的青睐。

在信息检索领域,SVD 是为了识别隐语义变量[9]而发展起来的。然而,由于大部分评分值的缺失,把 SVD 应用到 CF 领域的显式评分问题变得相对困难。当矩阵的信息不完整时,传统的 SVD 是不能被定义的。而且,简单地仅仅使用很少的已知信息将很容易导致过拟合现象。早期的研究依赖于填充方法[16, 26],即填充用户—物品评分矩阵的缺失值以使该矩阵变得稠密。然而,由于填充方法极大地增大了数据量,所以代价非常大。除此之外,不准确的填充也会使数据变得倾斜。因此,最近的研究[4, 6, 10, 17, 23, 24, 28]根据观察到的评分直接建模,并通过充分的正则化模型来避免过拟合。

在本节中,我们阐述了几种矩阵因子分解技术,这些技术的准确性随其复杂度增加而增加。我们首先讲述基本模型——SVD。然后讲述 SVD++ 模型,该模型整合了用户反馈的其他信息来提高预测准确性。最后我们考虑了这样的一个事实:用户的爱好或许会随着时间变化。同时,由于用户的爱好是不断变化的,他们甚至会重新选择他们的爱好。这导致了一个因子模型的产生,该模型考虑了时间效应信息用以更好地描述用户的行为。

#### 5.3.1 SVD<sup>⊖</sup>

矩阵因子分解模型把用户和物品两方面的信息映射到一个维度为  $f$  的联合隐语义空间中,因此用户—物品之间的交互作用被建模为该空间中的内积。这个隐语义空间试图通过描述物品和用户在各个因子上的特征来解释评分值,而这些因子是从用户反馈自动推断出的。例如,如果物品是电影,因子将会用来度量如喜剧或悲剧、情节的数量或者面向儿童的等级等这些明显的维度,以及如性格发展的深度或者“突变”等隐式维度,甚至是完全无法解释的维度。

相应地,每一个物品  $i$  都与一个  $f$  维向量  $q_i \in \mathbf{R}^f$  相关,每一个用户都与一个  $f$  维向量  $p_u \in \mathbf{R}^f$  相关。给定一个物品  $i$ ,  $q_i$  向量的维度值代表了该物品拥有这些因子的程度(例

⊖ SVD(Singular Value Decomposition)的想法是根据已有的评分情况,分析出评分者对各个因子的喜好程度以及电影包含各个因子的程度,最后再反过来根据分析结果预测评分。——译者注

如,某部电影的搞笑因子程度为5、而恐怖因子程度为1),其取值大小反映了物品拥有这些因子的积极或者消极程度。给定一个用户 $u$ ,向量 $p_u$ 的维度值代表了用户对这些因子的偏好程度(例如,某用户对搞笑因子的偏好程度为1,而对恐怖因子的偏好程度为0.1),同样地,这些值的大小反映了用户对这些因子的积极或者消极的评价。点积 $\odot q_i^T p_u$ 记录了用户和物品之间的交互,也就是用户对物品的总体兴趣度。加上之前提到的只依赖于用户或者物品的基准预测可以得到最终的评分。因此,评分通过下面的规则预测得到:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u \quad (5.2)$$

为了学习模型中的参数,也就是 $b_u, b_i, p_u, q_i$ ,我们可以最小化以下正则化的平方误差:

$$\min_{b_u, q_u, p_u, (u,i) \in K} \sum (r_{ui} - \mu - b_i - b_u - q_i^T p_u)^2 + \lambda_4 (b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2)$$

常量 $\lambda_4$ 控制了正则化程度,一般通过交叉验证来获得。最小化过程一般是通过随机梯度下降算法或者交替最小二乘法来实现的。

交替最小二乘法技术就是交替固定 $p_u$ 来计算 $q_i$ 和固定 $q_i$ 来计算 $p_u$ 。注意到当其中一个为常量时,最优化问题变成了二次的,就可以优化求解[2, 4]。

一种简单的随机梯度下降优化算法是由 Funk 推广[10]的并在其他研究者[17, 23, 24, 28]中得到成功的实践。该算法对训练数据中的所有评分做循环。对于给定的评分 $r_{ui}$ ,其预测评分记为 $\hat{r}_{ui}$ ,相关的预测误差记为 $e_{ui} \stackrel{\text{def}}{=} r_{ui} - \hat{r}_{ui}$ 。对于给定的训练样例 $r_{ui}$ ,我们通过朝着与梯度相反的方向移动来修正参数,如下所示:

- $b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda_4 \cdot b_u)$
- $b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda_4 \cdot b_i)$
- $q_i \leftarrow q_i + \gamma \cdot (e_{ui} \cdot p_u - \lambda_4 \cdot q_i)$
- $p_u \leftarrow p_u + \gamma \cdot (e_{ui} \cdot q_i - \lambda_4 \cdot p_u)$

当在 Netflix 数据上评估方法的时候,我们使用下面的参数值: $\gamma=0.005, \lambda_4=0.02$ 。下文,我们把这种方法叫作 SVD。

下面可以对该算法做些分析。通过为每类待学习参数精心选择学习率 $\gamma$ 和正则化因子 $\lambda$ 来提高准确度。因此,例如,可以对用户偏置、物品偏置和因子本身使用不同的学习率。该策略的集中使用在参考文献[29]中有详细描述。在本章中,当需要产生示范性的结果时,我们没有一直使用这个策略,而且在个别情况下,许多给定的常量并未调整到最优。

### 5.3.2 SVD++

考虑隐式反馈信息可以增加预测准确度,这些隐式反馈信息提供了用户爱好的额外指示。这对于那些提供了大量隐式反馈但是仅提供少量显式反馈的用户尤为重要。正如之前解释的那样,即使在独立的隐式反馈缺失的时候,我们也可以通过考虑用户评分的物品来得到用户兴趣的信息,而无须考虑这些物品的评分值。这样就出现了几种根据用户评分的物品来对用户的某个因子(用户对某个因子的喜好程度,比如,电影评分记录中用户对喜剧电影的喜爱程度因子)建模的方法[17, 23, 25]。本章重点关注 SVD++ 方法[17],可以证明这种方法能够提供比 SVD 方法更好的准确度。

$\odot$  记得两个向量 $x, y \in \mathbf{R}^f$ 之间的点积定义为 $x^T y = \sum_{k=1}^f x_k \cdot y_k$ 。

为了达到这个目的,我们增加了第二个物品因子集合,即为每一个物品  $i$  关联一个因子向量  $y_i \in \mathbf{R}^f$ 。这些新的物品因子向量根据用户评分的物品集合来描述用户的特征<sup>①</sup>。确切的模型如下:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T (p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j) \quad (5.3)$$

其中,集合  $R(u)$  包含用户  $u$  评分的所有物品。

现在,用户  $u$  的因子偏好程度被建模为  $p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j$ 。正如式(5.2)一样,我们使用用户  $u$  的因子偏好程度  $p_u$ ,这个向量从已知的显式评分记录学习得到。这个向量由  $|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j$  这一项做补充,这一项是从隐式反馈的角度出发的。由于  $y_j$  在 0 的附近取值(根据正则化),为了在观察值  $|R(u)|$  的整个范围内稳定其方差,我们用  $|R(u)|^{-\frac{1}{2}}$  来对其和做规范化。

模型的参数是通过采用随机梯度下降方法最小化相关联的正则化平方误差函数而取得的。我们在所有评分值已知的集合  $\mathcal{K}$  上做循环,计算:

- $b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda_5 \cdot b_u)$
- $b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda_5 \cdot b_i)$
- $q_i \leftarrow q_i + \gamma \cdot (e_{ui} \cdot (p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j) - \lambda_6 \cdot q_i)$
- $p_u \leftarrow p_u + \gamma \cdot (e_{ui} \cdot q_i - \lambda_6 \cdot p_u)$
- $\forall j \in R(u) : y_i \leftarrow y_i + \gamma \cdot (e_{ui} \cdot |R(u)|^{-\frac{1}{2}} \cdot q_i - \lambda_6 \cdot y_j)$

当在 Netflix 数据上评估方法时,我们使用下面的参数值:  $\gamma = 0.007$ ,  $\lambda_5 = 0.005$ ,  $\lambda_6 = 0.015$ 。而且最好是每一次迭代后减少步长(也就是  $\gamma$ )至原来的 0.9。迭代过程会持续 30 次迭代直至收敛。

通过使用额外的物品因子集合,可以把几种类型的隐式反馈同时引入模型中。例如,如果一个用户  $u$  对  $N^1(u)$  中的一些物品有某种类型的隐式偏好(如她租借了它们),对  $N^2(u)$  中的物品有另一种不同类型的偏好(如她检索过它们),我们可以使用下面的模型:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T (p_u + |N^1(u)|^{-\frac{1}{2}} \sum_{j \in N^1(u)} y_j^{(1)} + |N^2(u)|^{-\frac{1}{2}} \sum_{j \in N^2(u)} y_j^{(2)}) \quad (5.4)$$

每种隐式反馈信息的相对重要性将通过模型参数各自值的设定由算法自动学习得到。

### 5.3.3 时间敏感的因子模型

矩阵因子分解方法能很好地对时间效应建模,这样可以提高预测结果的准确度。通过把评分分解为不同的项,我们可以分别处理不同方面的时序影响。尤其是,我们可以定义下面随时间变化的因子:1)用户偏置  $b_u(t)$ ;2)物品偏置  $b_i(t)$ ;3)用户偏好  $p_u(t)$ 。另一方面,由于物品与人类不同,它在本质上是不可变的,所以我们也明确定义了物品的静态特征,原因在于我们并不希望物品有很大的时序变化。我们首先讨论基准预测内部的时序影响。

① 前面提到的 LFM 模型,即加入偏置项的 SVD 模型,并没有显式地考虑用户的历史行为对用户评分预测的影响,SVD++模型将用户历史评分的物品信息加入 LFM 模型中。——译者注

### 5.3.3.1 随时间变化的基准预测

基准预测的时序变化性主要体现在两个时间效应上面。第一个时间效应体现在物品的流行度或许随时间变化。例如，在一部新电影里某位演员的出现或许就会导致该电影的流行或过时。我们的模型中把物品偏置  $b_i$  看作时间的函数就能说明这一点。第二个时间效应体现在随着时间变化，用户或许会改变他们的基准评分。例如，一个过去倾向于对电影评分平均为4星的用户可能现在给出的平均评分为3星。这可能是由于用户评分标准的自然变化，也可能是用户对某件物品的评分跟最近他对其他物品的评分有关联，也可能是一个家庭里面的评分人会改变，也就是说并不都是由同一个人来评分的。因此，在我们的模型中，把用户偏置  $b_u$  也看作时间的函数。对一个时间敏感的基准预测来说，在  $t_{ui}$  天，用户  $u$  对物品  $i$  的评分可用下面公式来计算：

$$b_{ui} = \mu + b_u(t_{ui}) + b_i(t_{ui}) \quad (5.5)$$

式中， $b_u(\cdot)$  和  $b_i(\cdot)$  是随时间变化的实数函数。构造这些函数最好的方法必须要考虑如何参数化涉及的时序变化性。在电影评分数据集中我们做出的(构造这些函数的)选择向大家说明了一些典型的注意事项。

周期性变化的时间效应和相对瞬息万变的时间效应是有很大区别的。在电影评分案例中，一方面，我们不希望用户对电影的偏好程度每天都上下波动，而是在较长的一段时间后才发生变化。另一方面，我们注意到用户的影响每天都会变，这也反映了消费者行为不是持续不变的本质特性。对用户偏置的建模需要一个较细的时间粒度，而对与物品相关的偏置建模时选择一个相对较粗的时间粒度就可以了。

我们首先确定如何选择随时间变化的物品偏置  $b_i(t)$ 。我们发现完全可以把物品偏置分割为不同的时间段来计算，而每一个时间段都用一个常数表示物品偏置。把时间轴划分到不同的时间段时，既希望时间粒度较细(时间段较短)，又需要每个时间段包含足够的评分记录(时间段较长)，这就需要权衡。对电影评分数据来说，在相同准确度条件下，可选择的时间段大小的范围非常宽泛。在具体的实现中，每一个时间段对应大约连续十周的评分数据，需要30个时间段来跨度数据集中的所有天数。天数  $t$  关联着一个整数  $\text{Bin}(t)$ ，(在数据集中，取值为1~30)，于是电影偏置就被分为一个固定部分和一个随时间变化的部分。

$$b_i(t) = b_i + b_{i, \text{Bin}(t)} \quad (5.6)$$

尽管把参数分时间段取值在物品偏置上很好，但是很难推广到用户偏置的计算上。一方面，我们希望对用户采用精细的时间粒度，用以发现非常短时间的的时间效应。另一方面，我们又不能期望每个用户都会有足够的评分数据，用以对独立的时间段做出可靠的估计。我们可以考虑用不同的函数形式来参数化随时间变化的用户行为，当然每个函数形式的复杂度和准确度都不同。

一个简单的建模选择是使用一个线性函数来模拟用户偏置可能的渐变过程。对每个用户  $u$ ，定义该用户评分日期的均值为  $t_u$ 。现在，若用户  $u$  在  $t$  天的时候评价了一部电影，则与该评分相关的时间偏置定义为

$$\text{dev}_u(t) = \text{sign}(t - t_u) \cdot |t - t_u|^\beta$$

该公式中， $|t - t_u|$  是日期  $t$  和  $t_u$  间隔的天数。我们通过交叉验证来设置  $\beta$  的值，这里采取  $\beta = 0.4$ 。我们为每个用户引入一个单独的新参数  $\alpha_u$ ，这样便得到了第一个与时间相关的用户偏置：

$$b_u^{(1)}(t) = b_u + \alpha_u \cdot \text{dev}_u(t) \quad (5.7)$$

这个用来近似随时间变化的用户行为的简单线性模型需要为每一个用户  $u$  学习两个参数:  $\beta_u$  和  $\alpha_u$ 。

我们也可以采用曲线来进行更灵活的参数化。假设用户  $u$  有  $n_u$  条评分记录。指定  $k_u$  个时间点,  $\{t_1^u, \dots, t_{k_u}^u\}$ , 这些时间点把用户评分记录日期进行了均匀的划分, 并且作为核控制着下面的函数:

$$b_u^{(2)}(t) = b_u + \frac{\sum_{l=1}^{k_u} e^{-\sigma|t-t_l^u|} b_{t_l^u}^u}{\sum_{l=1}^{k_u} e^{-\sigma|t-t_l^u|}} \quad (5.8)$$

参数  $b_{t_l^u}^u$  与控制点(也叫核)相关, 它们从数据中自动学习得到。这样用户偏置就由这些参数的时间加权组合组成。控制点的个数  $k_u$  用于调和算法灵活性和计算效率。在这里设  $k_u = n_u^{0.25}$ , 使其随着可用的评分记录增加而增加。常量  $\sigma$  决定了曲线的平滑度, 通过交叉验证将其值设为  $\sigma = 0.3$ 。

目前为止, 我们讨论了对用户偏置建模时采用平滑函数, 这些平滑函数非常适合模拟逐渐观念转变的时间效应情形。然而, 在很多应用中, 有很多在某一天或某段时间会发生瞬间变化的突变情形。例如, 在电影评分数据集中, 我们发现在某一天某个用户给出的众多评分值往往集中在一个单一的值。这样的结果跨度只在某一天之内。这个结果或许反映了用户当天的心情, 或许是受到每个其他用户评分的影响, 或许是用户评分标准的真实变化。为了处理这样短时间内存在的影响, 我们为每一个用户和每一天指定了一个参数用以反映特定天的变化。这个参数记为  $b_{u,t}$ 。注意在一些应用中, 真正采用的基本时间单位可以比一天短或长一些。

在 Netflix 电影评分数据中, 一个用户平均在 40 个不同的日期对电影评分。因此, 获得参数  $b_{u,t}$  时, 平均需要 40 个参数来描述每个用户偏置。由于  $b_{u,t}$  缺失了所有跨度超过一天的信息, 所以把任务  $b_{u,t}$  作为单独的一个变量并不能充分地处理用户偏置。因此, 把它作为前面描述过的模式中的一个附加部分。于是, 时间线性模型(5.7)变成:

$$b_u^{(3)}(t) = b_u + \alpha_u \cdot \text{dev}_u(t) + b_{u,t} \quad (5.9)$$

相类似, 基于曲线的模型变成下面的模型:

$$b_u^{(4)}(t) = b_u + \frac{\sum_{l=1}^{k_u} e^{-\sigma|t-t_l^u|} b_{t_l^u}^u}{\sum_{l=1}^{k_u} e^{-\sigma|t-t_l^u|}} + b_{u,t} \quad (5.10)$$

仅靠基准预测并不能产生个性化推荐, 原因在于它忽略了用户和物品之间的所有交互。在某种意义上, 它只是抓住了与建立推荐不是很相关的那部分数据。然而, 为了比较与时间相关的、不同类别的用户偏置的优缺点, 我们在单个预测器上比较它们的准确度。为了学习模型中涉及的参数, 我们用随机梯度下降算法来最小化相关的正则化平方误差。例如, 在真实的实现中, 我们采用规则(5.9)来对随时间变化的用户偏置建模, 这样得到下面的基准预测器:

$$b_{ui} = \mu + b_u + \alpha_u \cdot \text{dev}_u(t_{ui}) + b_{u,t_{ui}} + b_i + b_{i, \text{Bin}(t_{ui})} \quad (5.11)$$

为了学习涉及的参数  $b_u$ 、 $\alpha_u$ 、 $b_{u,t}$  和  $b_i$ 、 $b_{i, \text{Bin}(t)}$ , 这里等价于解决下面的最小化问题:

$$\min \sum_{(u,i) \in K} (r_{ui} - \mu - b_u - \alpha_u \text{dev}_u(t_{ui}) - b_{u,t_{ui}} - b_i - b_{i, \text{Bin}(t_{ui})})^2 + \lambda_7 (b_u^2 + \alpha_u^2 + b_{u,t_{ui}}^2 + b_i^2 + b_{i, \text{Bin}(t_{ui})}^2)$$



这里，第一项试图构造与已知评分拟合得最好的参数。正则化项  $\lambda_7(b_u^2 + \dots)$  通过对参数的大小进行惩罚来避免过拟合现象，这些参数的初始值设为 0。学习过程是通过随机梯度下降算法在 20~30 次迭代之后完成的，其中  $\lambda_7 = 0.01$ 。

表 5.1 比较之前提出的几种基准预测器解读数据中信息的能力。与往常一样，捕获的信息是由根均方差(RMSE)来度量的。在此提醒一下，由于测试案例在时间上比训练样例出现的晚，所以预测经常涉及关于时间的外推法。我们可以这样表示这些基准预测器：

表 5.1 电影和用户主要影响的基准预测器之间的比较。随着时序建模越来越准确，预测准确性也随之提高(RMSE 变小)

模型	static	mov	linear	spline	linear+	spline+
RMSE	0.979 9	0.977 1	0.973 1	0.971 4	0.960 5	0.960 3

- 静态模型，不考虑时间效应： $b_{ui} = \mu + b_u + b_i$ 。
- Mov 模型，只考虑与电影相关的时间效应： $b_{ui} = \mu + b_u + b_i + b_{i, \text{Bin}(t_{ui})}$ 。
- 线性模型，考虑用户偏置的线性模型： $b_{ui} = \mu + b_u + \alpha_u \cdot \text{dev}_u(t_{ui}) + b_i + b_{i, \text{Bin}(t_{ui})}$ 。
- 样条曲线模型，考虑用户偏置的样条曲线模型： $b_{ui} = \mu + b_u + \frac{\sum_{l=1}^{k_u} e^{-\sigma} |t_{ui} - t_l^u| b_{t_l^u}^u}{\sum_{l=1}^{k_u} e^{-\sigma} |t_{ui} - t_l^u|} + b_i + b_{i, \text{Bin}(t_{ui})}$ 。
- linear+模型，考虑用户偏置和单天效应的线性模型： $b_{ui} = \mu + b_u + \alpha_u \cdot \text{dev}_u(t_{ui}) + b_i + b_{u, t_{ui}} + b_{i, \text{Bin}(t_{ui})}$ 。
- spline+模型，考虑用户偏置和单天效应的样条曲线模型： $b_{ui} = \mu + b_u + \frac{\sum_{l=1}^{k_u} e^{-\sigma} |t_{ui} - d_l| b_{t_l^u}^u}{\sum_{l=1}^{k_u} e^{-\sigma} |t_{ui} - t_l^u|} + b_i + b_{u, t_{ui}} + b_{i, \text{Bin}(t_{ui})}$ 。

表 5.1 显示，尽管考虑电影评分数据中电影的时间效应提高了预测准确度(把 RMSE 从 0.9799 降低到 0.9771)，但相比之下，用户偏置的变化影响更大。跟线性模型相比，建模时使用样条曲线增加了额外的灵活性，因此提高了预测准确度。然而，由每天参数捕获的用户偏置的突变才是最重要的。事实上，当考虑这些突变时，线性模型(linear+)和样条曲线模型(spline+)的区别几乎消失了。

除了捕捉目前描述的时间效应，可以通过相同的方法捕获更多的其他效应。一个主要的案例是捕获季节效应。例如，一些产品或许在特定季节或者临近某些假期时才会流行。相类似，电视或者电台节目在一天的不同时间段内才会流行(也就是“分时段”)。季节效应也可以在用户端出现。例如，用户或许在周六日和工作日有不同的心态或者购买模式。对这样的效应建模的一种方式是用一个参数把时间周期和用户或物品结合起来。这样，式(5.6)所示的物品偏置变为：

$$b_i(t) = b_i + b_{i, \text{Bin}(t)} + b_{i, \text{period}(t)}$$

例如，如果我们试图捕获物品偏置在一年不同季节内的变化，则  $\text{period}(t) \in \{\text{fall, winter, spring, summer}\}$ 。相类似，我们或许可以通过把式(5.9)修改为下面的公式来对周期性的用户影响建模：

$$b_u(t) = b_u + \alpha_u \cdot \text{dev}_u(t) + b_{u,t} + b_{u,\text{period}(t)}$$

然而, 尚未发现季节效应在电影评分数据集中有重要的预测能力, 因此我们报告的结果并没有包含这些季节效应。

另一种基准预测器范围内的时间效应是与用户评分标准的变化相关的。尽管  $b_i(t)$  是一个与用户无关的、在时刻  $t$  关于物品  $i$  价值的度量, 用户却倾向于对这样的一个度量做出不同的回应。例如, 不同的用户有不同的评分标准, 并且是同一个用户的评分标准也会随着时间改变。于是, 电影偏置的原始数值就不是完全与用户无关的了。为了解决这个问题, 我们为基准预测器增加一个与时间相关的扩展特征, 记为  $c_u(t)$ 。因此, 基准预测器式(5.11)变成:

$$b_{ui} = \mu + b_u + \alpha_u \cdot \text{dev}_u(t_{ui}) + b_{u,t_{ui}} + (b_i + b_{i,\text{Bin}(t_{ui})}) \cdot c_u(t_{ui}) \quad (5.12)$$

上面讨论的用于实现  $b_u(t)$  的方法都可以用来实现  $c_u(t)$ 。我们设定一个随天数变化的单独参数, 得到  $c_u(t) = c_u + c_{u,t}$ 。与往常一样,  $c_u$  是  $c_u(t)$  的稳定部分, 而  $c_{u,t}$  代表了特定天的变化。把  $c_u(t)$  当作一个乘法因子增加到基准预测器可以把 RMSE 降低到 0.9555。有趣的是, 只捕获主要影响而完全忽略用户—物品交互作用的基本模型, 与商业化的 Netflix Cinematch 推荐系统一样, 都可以用来解释大部分的数据变化。在相同的测试集上, Netflix Cinematch 推荐系统报告的 RMSE 为 0.9514[5]。

### 5.3.3.2 随时间变化的因子模型

在 5.3.3.1 节中, 我们讨论了时间因子对基准预测器的影响。然而, 正如之前提到的那样, 时间效应不仅对基准预测器有影响, 也对用户的偏好有影响, 从而影响用户和物品之间的交互作用。用户的偏好随着时间变化。例如, “心理惊悚片”类型的电影迷或许一年后变成“罪案片”的电影迷。相类似, 人们会改变他们对特定导演和演员的看法。这种演进通过把用户因子(向量  $p_u$ )作为时间的函数来建模。再一次, 我们需要在面临用户评分的内置稀疏性的困境下, 以每天这样的精细时间粒度来对这些变化建模。事实上, 这些时间效应是最难捕获到的, 因为用户偏好并不像主要影响(用户偏置)那样明显, 而是被分割成了许多因素。

与处理用户偏置的方式相似, 我们对用户偏好的每一个组成部分进行建模, 即  $p_u(t)^T = (p_{u,t}(t), \dots, p_{u,f}(t))$ 。在电影评分数据集中, 我们发现以式(5.9)来建模是有效的, 如下所示:

$$p_{uk}(t) = p_{uk} + \alpha_{uk} \cdot \text{dev}_u(t) + p_{u,k,t} \quad k = 1, \dots, f \quad (5.13)$$

这里,  $p_{uk}$  代表因子的不变部分,  $\alpha_{uk} \cdot \text{dev}_u(t)$  是对可能随时间线性变化的那部分的近似,  $p_{u,k,t}$  代表了非常局部性的、与特定天相关的变化。

这时, 我们可以把所有的碎片拼凑起来并通过整合这些随时间变化的参数来扩展 SVD 因子模型。得到的扩展模型记为 timeSVD++, 且预测规则如下所示:

$$\hat{r}_{ui} = \mu + b_{i(t_{ui})} + b_{u(t_{ui})} + q_i^T (p_{u(t_{ui})} + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j) \quad (5.14)$$

随时间变化的参数  $b_i(t)$ 、 $b_u(t)$  和  $p_u(t)$  在式(5.6)、式(5.9)和式(5.13)中有精确的定义。学习过程通过使用随机梯度下降算法来最小化数据集上相关的平方误差函数来完成。整个过程与原始的 SVD++ 算法类似。每一次迭代的时间复杂度仍然与输入大小呈线性关系, 而运行时间大概是 SVD++ 算法的两倍, 原因是更新时序参数需要额外的花费。重要的是, 收敛速度并没有受时序参数化的影响, 大概经历 30 次迭代后该算法就会收敛。

### 5.3.4 比较

表 5.2 比较了本节讨论的三种算法的结果。首先是 SVD 方法，最原始的矩阵分解算法。其次是 SVD++ 方法，该方法在 SVD 的基础上整合了一种隐式反馈，因此提高了预测准确度。最后是 timeSVD++，该模型考虑了时间效应。这三种方法在一系列分解维度 ( $f$ ) 进行了比较。三种方法的预测准确度都随着因子维度数目的增加而提高，原因是因子维度个数越高，它们表达电影—用户的复杂交互的能力就越强。注意到 SVD++ 模型中参数的数量跟 SVD 是相当的。这是因为 SVD++ 模型只是加入了物品方面的影响，而数据集的复杂度是由更大的用户集来决定的。另一方面，timeSVD++ 模型需要更多的参数，因为它精确表示了每一个用户因子。在 SVD++ 模型中增加隐式反馈也可以提高在电影评分数据集上的预测准确度。但是，与 SVD++ 模型相比，timeSVD++ 模型带来的预测准确度的提高一直都很明显。我们尚未发现文献中的哪个单独的算法能够有如此高的准确度。捕获时间效应的重要性进一步体现了这样的一个事实：因子维度为 10 的 timeSVD++ 模型已经比因子维度为 200 的 SVD 模型性能好。相类似，因子维度为 20 的 timeSVD++ 模型已经比因子维度为 200 的 SVD++ 模型性能好。

表 5.2 三种因子模型的比较：预测准确性是在不同因子维度 ( $f$ ) 下由 RMSE 度量的 (RMSE 越小，证明预测准确性越高)。对于所有的模型，准确度随因子维度的个数增加而提高。SVD++ 模型在 SVD 模型的基础上整合了隐式反馈，因子提高了准确度。timeSVD 在 SVD++ 的基础上增加了数据中时间效应的影响，因此进一步增加了准确度

模型	$f=10$	$f=20$	$f=50$	$f=100$	$f=200$
SVD	0.914 0	0.907 4	0.904 6	0.902 5	0.900 9
SVD++	0.913 1	0.903 2	0.895 2	0.892 4	0.891 1
timeSVD++	0.897 1	0.889 1	0.882 4	0.880 5	0.879 9

#### 5.3.4.1 预测未来日期

我们的模型包含了特定天的参数。一个明显的问题就是如何使用这些模型来预测未来的评分，即对我们还不能训练特定天的参数的这些日期进行评分。一个简单的答案是对于这些未来(未训练)的日期，特定日期的参数应该取默认值。也就是，对于式 (5.12)， $c_u(t_w)$  设为  $c_u$ ， $b_{u,t_w}$  设为 0。然而，有人会问，如果我们不能使用这些特定天的参数来预测未来的日期，那这些参数在什么地方才有用呢？毕竟，预测只有关乎未来才是有用的。为了使这个问题更加清晰，我们需要提到这样一个事实：在 Netflix 数据集中包括了很多用户在某些日期只有一条评分记录的数据，因此，特定日期的参数不能使用。

为了回答这个问题，需要注意时序化建模并没有试图捕获未来的变化。我们的时序化建模试图做的是捕获瞬息万变的动态时序，这些动态时序对用户过去的反馈有很大的影响。当我们识别出这些时间效应时，应该对它们进行向下调整，以便我们能够对更加持久化的信号建模。这样，模型就可以更好地捕获数据的长期特征，而使用精心设计的参数来表示短期的波动。例如，如果一个用户某天给出了很多比平时评分要高的评分，我们的模型将会对这些评分给予一定的折扣，因为考虑到这可能是用户在特定天的好心情造成的，而这并不能反映该用户长期的行为。这样，特定天的参数完成了一种数据清理工作，而这种数据清理提高了对未来日期的预测准确度。

### 5.3.5 总结

矩阵分解模型的基本形式描述了物品和用户两方面的特征,而这是通过由物品的评分模式推导出的向量因子实现的。物品和用户因子间的高度一致性才会导致一个物品被推荐给一个用户。这些方法的预测准确度要优于已经发表的其他协同过滤技术。同时,这些方法提供了一个内存有效的压缩模型,该模型训练起来相对容易。这些优点,加上基于梯度下降算法的矩阵分解模型实现起来很容易,使得该方法成为 Netflix 有奖比赛中使用的方法之一。

这些技术在处理数据多个关键方面的能力使得它们在实际应用中更加方便。首先,是整合多种形式的用户反馈的能力。我们可以观察用户其他相关的行为来更好地预测该用户的评分,如用户的购买或者浏览历史记录。提出的 SVD++ 模型利用了多种形式的用户反馈来提高用户评分的预测准确度。

另一个重要方面是时间效应,这些时间效应反映了用户随时间变化的爱好。每一个用户和产品在其特征上都会潜在地经历一系列不同的变化。在随时间变化的数据中,仅仅靠旧实例的衰减不能充分地识别出公共的行为模式。我们采用的解决方法是在整个时期对时间效应建模,这样就可以智能地把瞬态因子和持续性的因子分离开。包含时间效应的模型被证明在提高预测质量方面比算法上的增强更有用。

## 5.4 基于邻域模型

协同过滤领域最常见的方法是基于邻域的模型。第 4 章对该方法进行了广义概括。其最原始形式是基于用户的,该形式是早期所有 CF 系统采用的共同方法,参见文献[14]来得到更好的分析。基于用户的方法是基于一群志趣相同的用户的评分记录来估计未知的评分。

随后,一种类似的基于物品的方法[20, 27]流行起来。在这些方法中,我们使用同一个用户在相似物品上的评分来估计未知的评分。由于基于物品的方法具有更好的可扩展性并提高了准确度,所以该方法在很多种场景中都适用[2, 27, 28]。除此之外,基于物品的方法能更好地解释预测背后的原因。这是因为用户对他们之前喜爱过的物品比较熟悉,但是他们却不认识这些所谓的志趣相同的人。我们主要关注基于物品的方法,但是同样的技术可以在基于用户的方法中直接使用,可以参考 5.5.2.2 节。

一般来说,隐语义模型在描述数据的各种方面具有很强的表达能力。因此它们的预测结果一般比基于邻域的模型要好。然而,大多数文献和商业系统(例如,Amazon[20]和 Tivo[1]的推荐系统)都是以基于邻域的模型为基础的。基于邻域的模型之所以如此普遍部分,是因为它们相对简单。然而,现实生活中坚持采用这种模型有更重要的原因。首先基于邻域的模型提供了推荐背后的直观解释,这种解释不仅提高了准确性,同时增强了用户体验(解释推荐系统的更多内容在本书第 15 章中有更多叙述)。其次,基于邻域的模型能够根据一个新进入系统的用户反馈立即提供推荐。

本节的结构如下。首先,描述如何估计两个物品之间的相似度,这是大多数基于邻域的模型的基础。其次,我们转到广泛使用的基于相似度的邻域方法,该方法构成了相似度权重的一个直接应用。我们指出这种基于相似度的方法的一些特定的限制。最后,在 5.4.3 节中,给出解决这些问题的建议方法,通过这种方法,可以以计算时间上微小的增加为代价来提高预测准确度。

### 5.4.1 相似度量

基于物品的方法的核心是物品之间的相似度量。一般情况下相似度的度量是基于皮尔逊相关系数  $\rho_{ij}$ ，该相关系数度量了用户对物品  $i$  和物品  $j$  进行评分的相似性趋势。由于许多评分未知，一些物品或许只有几个公共的评分用户。经验相关系数  $\hat{\rho}_{ij}$  仅仅是基于共同的用户支持。我们建议用基准预测器的残差 ( $b_{ui}$ ，见 5.2.1 节) 来补偿特定用户和特定物品的偏差。因此，近似的相关系数如下所示：

$$\hat{\rho}_{ij} = \frac{\sum_{u \in U(i,j)} (r_{ui} - b_{ui})(r_{uj} - b_{uj})}{\sqrt{\sum_{u \in U(i,j)} (r_{ui} - b_{ui})^2 \cdot \sum_{u \in U(i,j)} (r_{uj} - b_{uj})^2}} \quad (5.15)$$

集合  $U(i, j)$  包含了同时对物品  $i$  和  $j$  评分的用户。

由于基于更大的用户支持的相关系数估计值更加可靠，所以我们可以使用一种近似的相似度量，记为  $s_{ij}$ ，这是一个相关系数的收缩值，其形式如下：

$$s_{ij} \stackrel{\text{def}}{=} \frac{n_{ij} - 1}{n_{ij} - 1 + \lambda_8} \rho_{ij} \quad (5.16)$$

变量  $n_{ij} = |U(i, j)|$  记为同时对物品  $i$  和  $j$  评分的用户数量。 $\lambda_8$  的典型值可取 100。

这样收缩是从贝叶斯角度出发的，参考 2.6 节中 Gelman 等的文献[11]。假设真实的  $\rho_{ij}$  是服从正态分布的独立随机变量，即给定  $\tau^2$ ：

$$\rho_{ij} \sim N(0, \tau^2)$$

如果  $b_{ui}$  考虑到用户和物品对各自均值的偏离，均值被调整为 0。同时，给定  $\sigma_{ij}^2$ ，假设：

$$\hat{\rho}_{ij} | \rho_{ij} \sim N(\rho_{ij}, \sigma_{ij}^2)$$

我们通过其后验均值来估计  $\rho_{ij}$ ： $E(\rho_{ij} | \hat{\rho}_{ij}) = \frac{\tau^2 \hat{\rho}_{ij}}{\tau^2 + \sigma_{ij}^2}$ 。经验估计值  $\hat{\rho}_{ij}$  在趋于 0 时缩小了一个比例  $\sigma_{ij}^2 / (\tau^2 + \sigma_{ij}^2)$ 。

式(5.16)通过等式  $\sigma_{ij}^2 = 1/(n_{ij} - 1)$  来逼近相关系数的方差，此时  $\rho_{ij}$  的值接近于 0。

注意文献[27, 28]提出了相似度度量的其他方法。

### 5.4.2 基于相似度的插值

这里描述基于邻域建模的最流行方法，显然该方法对一般意义上的 CF 也适用。我们的目标是预测  $r_{ui}$ ——尚未观察到的用户  $u$  对物品  $i$  的评分。使用这种相似度量，我们可以找到用户评分的、与物品  $i$  最相似的  $k$  个物品。这个由  $k$  个近邻物品组成的集合记为  $S^k(i; u)$ 。 $r_{ui}$  的预测值取用户对这些紧邻物品评分的加权平均，同时用户和物品在基准预测器的影响下做调整，得到的预测规则如下：

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in S^k(i,u)} s_{ij} (r_{uj} - b_{uj})}{\sum_{j \in S^k(i,u)} s_{ij}} \quad (5.17)$$

注意相似度的双重用途：一是识别出最近的近邻物品；二是作为式(5.17)中的插值权重。

有时，除了直接把相似度权重作为插值系数，我们也可以转换这些权重来得到更好的



结果。例如，我们已经在一些数据集中发现取基于关联的相似度的平方值很有用。这样得

到的规则： $\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in S^k(i, u)} s_{ij}^2 (r_{uj} - b_{uj})}{\sum_{j \in S^k(i, u)} s_{ij}^2}$ 。Toscher 等在文献[31]中讨论了这些权重更复杂的

的转换形式。

基于相似度的方法之所以变得这么流行，是因为它们很直观，并且实现起来相对简单。它们还具有下面两个很有用的特性。

- **可解释性。**解释自动推荐系统的重要性在文献[13, 30]中得到广泛认可，也可以参考第 15 章。用户希望系统给出预测的原因，而不是像“黑盒”一样仅仅展示推荐列表。解释不仅丰富了用户体验，而且可以鼓励用户与系统交互、修正与主观印象违背的内容来提高长期的准确性。基于邻域的框架可以识别出用户过去的哪个行为对计算出来的预测影响最大。
- **新的评分。**基于物品的邻域模型能够在用户输入新的评分后立即给出更新过的推荐结果。这包括一旦用户对系统提供反馈时就立即处理新的用户，而不需要重新训练模型以及估计新的参数。这里假设物品之间的关系( $s_{ij}$  的值)是稳定的，并不是每天都变。注意对于新进入系统的物品，我们确实需要学习新的参数。有趣的是，用户和物品间的这种非对称性在常见的应用中配合得很好：系统需要对新进入系统的用户(或者老用户的新评分)立即做出推荐，因为这些用户期望有质量的服务。另一方面，物品进入系统后，等待特定时间再把它们推荐给用户也是合情合理的。

然而，基于邻域的标准模型也面临下面的问题：

- 直接定义插值权重的相似度函数( $s_{ij}$ )是任意取的。不同的 CF 算法使用多少有点不同的相似度度量，用以量化用户相似度或者物品相似度这一难以捉摸的概念。假设一个特定的物品的评分可由其近邻物品的一个子集完美地预测出来。这种情况下，我们想让这个预测子集能得到所有的权重，但是这对于像使用皮尔逊相关系数这样的有界相似度得分是不可能的。
- 之前基于邻域的方法没有考虑近邻物品之间的相互作用。物品  $i$  和其近邻物品  $j$  的每一个相似度在计算时是完全独立于集合  $S^k(i; u)$  的内容和其他相似度  $s_{il}$  的，其中  $l \in S^k(i; u) - \{j\}$ 。例如，假设物品是电影，邻域集合包含了相互之间高度相关的三部电影(例如，“指环王 1~3”续集)。在决定插值权重时，如果一个算法忽略了这三部电影的相似性，则最终或许会把该组物品提供信息实际计算三次。
- 根据定义，插值权重的和为 1，这或许会导致过拟合。假设一个物品没有被某个特定用户评分的有用的近邻。这种情况下，最好忽略掉邻域信息，直接根据更具健壮性的基准预测器进行预测即可。然而，标准邻域公式使用了这些不提供有用信息的近邻物品评分的加权平均。
- 如果近邻间物品的评分变化太大，基于邻域的方法工作得不好。

其中一些问题可以在一定程度上得到解决，然而其他问题在基本框架内很难解决。例如，第三个问题，处理权重之和为 1 的约束时，可以使用下面的预测规则缓解这个约束：

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in S^k(i, u)} s_{ij} (r_{uj} - b_{uj})}{\lambda_0 + \sum_{j \in S^k(i, u)} s_{ij}} \quad (5.18)$$

常量  $\lambda_9$  代表了当近邻信息很少时对基于邻域的部分的惩罚, 例如, 当  $\sum_{j \in S^k(i; u)} s_{ij} \ll \lambda_9$  时。确实, 我们已经发现, 为  $\lambda_9$  设定一个合适的值能够在式(5.17)基础上提高准确性。但整个框架并没有被一种正式的模型所证明。因此, 我们试图用一种更加基础的方法来得到更好的结果, 下面讨论这种方法。

### 5.4.3 联合派生插值权重

本节描述一种更加准确的邻域模型, 该模型克服了上面讨论的所有困难, 同时又保持了基于物品的模型的已知优点。与上面一样, 我们为每一个预测使用相似度量来定义近邻。然而, 我们寻找最优插值时并没有考虑相似度量度的值。

给定一个近邻集合  $S^k(i; u)$ , 我们需要计算插值权重  $\{\theta_{ij}^u | j \in S^k(i; u)\}$ , 通过这些权重可以得到下面形式的最好预测规则:

$$\hat{r}_{ui} = b_{ui} + \sum_{j \in S^k(i; u)} \theta_{ij}^u (r_{uj} - b_{uj}) \quad (5.19)$$

$k$ (近邻的个数)的典型取值范围为 20~50, 参考文献[2]。在本节中, 我们假设基准预测器已经被移除。因此, 我们为残差引入一个符号,  $z_{ui} \stackrel{\text{def}}{=} r_{ui} - b_{ui}$ 。为了使符号方便使用, 我们假设集合  $S^k(i; u)$  中物品的索引为 1,  $\dots$ ,  $k$ 。

寻找直接在预测规则(5.19)使用的插值权重的正式的计算方法。正如之前解释过的那样, 推导所有插值权重时同时考虑近邻物品之间的相互依赖是很重要的。我们通过定义一个相应的最优化问题来达到这些目的。

#### 5.4.3.1 形式化模型

首先, 考虑一种假设密集的情形, 在这种情形里, 除了  $u$  的所有用户都同时对物品  $i$  和它在集合  $S^k(i; u)$  中的所有近邻物品进行评分。这种情形下, 我们可以通过对物品  $i$  和它的近邻物品之间的关联进行建模来学习这些插值权重, 建模是通过解决下面的最小二乘法问题完成的:

$$\min_{\theta^u} \sum_{v \neq u} (z_{vi} - \sum_{j \in S^k(i; u)} \theta_{ij}^u z_{vj})^2 \quad (5.20)$$

注意, 这里唯一的未知量是  $\theta_{ij}^u$ 。该最小二乘法问题(5.20)的最优解是通过转化为求解一个线性方程组而得到的。从统计学的角度看, 这等价于在  $z_{vi}(j)$  上对  $z_{vi}$  进行无截距回归的结果。特别地, 最优权重由下面的等式给出:

$$Aw = b \quad (5.21)$$

这里,  $w \in \mathbf{R}^k$  是一个未知向量, 其中  $w_j$  代表要寻找的系数  $\theta_{ij}^u$ 。  $A$  是一个  $k \times k$  矩阵, 其中:

$$A_{jl} = \sum_{v \neq u} z_{vj} z_{vl} \quad (5.22)$$

相类似, 向量  $b \in \mathbf{R}^k$  定义如下:

$$b_j = \sum_{v \neq u} z_{vj} z_{vi} \quad (5.23)$$

对于一个稀疏的评分矩阵, 有可能同时对物品  $i$  和其邻域内的物品进行评分的用户很少。因此, 把式(5.22)和式(5.23)所示的  $A$  和  $b$  仅仅基于那些有完整数据的用户的数据去计算是不明智的。即使有足够多的包含完整数据的用户来保证矩阵  $A$  是非奇异的, 那样的估计也会忽略同一个用户评分之间的成对关系的大部分信息。然而, 通过在给定的成对关

系的支持上求平均, 我们仍然能够估计  $A$  和  $b$ , 与上面相似的常量。这样得到下面的改进公式:

$$\bar{A}_{jl} = \frac{\sum_{v \in U(j,l)} z_{vj} z_{vl}}{|U(j,l)|} \quad (5.24)$$

$$\bar{b}_j = \frac{\sum_{v \in U(i,j)} z_{vj} z_{vi}}{|U(i,j)|} \quad (5.25)$$

在此提示,  $U(j, l)$  是同时对物品  $j$  和  $l$  评分的用户的集合。

这仍然不能够克服稀疏问题。 $\bar{A}_{jl}$  或  $\bar{b}_j$  的元素可能会因为被用于计算平均值的用户组的排序而改变<sup>①</sup>。正如之前讨论的那样, 基于相对低支持度 ( $|U(j, l)|$  的值较小) 的均值可以通过收缩到一个共同值来改进。尤其是, 我们计算一个基准值, 该基准值是通过所有可能的  $\bar{A}_{jl}$  的值的平均值来定义的。定义这个基准值为  $\text{avg}$ , 它的精确定义在 5.4.3.2 节中描述。于是, 定义相应的  $k \times k$  矩阵  $\hat{A}$  和向量  $\hat{b} \in \mathbf{R}^k$ :

$$\hat{A}_{jl} = \frac{|U(j,l)| \cdot \bar{A}_{jl} + \beta \cdot \text{avg}}{|U(j,l)| + \beta} \quad (5.26)$$

$$\hat{b}_j = \frac{|U(i,j)| \cdot \bar{b}_j + \beta \cdot \text{avg}}{|U(i,j)| + \beta} \quad (5.27)$$

参数  $\beta$  控制了收缩的程度。一个典型的取值为  $\beta=500$ 。

假设  $A$  和  $b$  的最佳估计值分别为  $\hat{A}$  和  $\hat{b}$ 。因此, 修改式 (5.21) 以便将插值权重定义为下面线性方程组的解:

$$\hat{A}w = \hat{b} \quad (5.28)$$

得到的插值权重在式 (5.19) 中使用以预测  $r_w$ 。

这个方法解决了 5.4.2 节提出的四个问题。第一, 插值权重直接从评分中推导得到, 而不是基于任何相似度度量。第二, 插值权重公式显式地考虑了近邻之间的联系。第三, 权重的和并没有被约束为 1。如果一个用户 (或物品) 的近邻信息很少, 则估计权重会很小。第四, 该方法随着物品的均值或方差的变化自动地调整。

#### 5.4.3.2 计算时的问题

基于物品的邻域方法的有效计算需要提前计算与每个物品对相关的特定值, 这样可以达到快速检索的目的。首先, 我们需要通过 5.4.1 节解释的那样提前计算所有的  $s_{ij}$  的值, 从而快速访问到所有的基于物品的相似度。

其次, 我们提前计算  $\hat{A}$  和  $\hat{b}$  的所有项。为了达到这个目的, 对于每两个物品  $i$  和  $j$ , 计算:

$$\bar{A}_{ij} = \frac{\sum_{v \in U(i,j)} z_{vi} z_{vj}}{|U(i,j)|}$$

然后, 前面提到的式 (5.26) 和式 (5.27) 中的基准值  $\text{avg}$ , 取值为提前计算的  $n \times n$  矩阵  $\hat{A}$  的所有项的平均值。事实上, 我们推荐使用两种不同的基准值, 一种是取矩阵  $\hat{A}$  的非对角项的平均值, 一种是取矩阵  $\hat{A}$  相对较大的对角项的平均值, 这些对角值的均值之所以较大是因为它们只对非负项求和。最后, 使用  $\text{avg}$  的近似值, 从式 (5.26) 所示的矩阵  $\hat{A}$  推导

① 根据前面所述的 top  $k$  可知。——译者注

出一个完全  $n \times n$  矩阵  $\hat{A}$ 。这里, 当推导  $\hat{A}$  的非对角项时, 我们使用非对角项的均值, 而推导  $\hat{A}$  的对角项时, 我们使用对角项的均值。

由于对称性, 我们只存储满足  $i \geq j$  条件的  $s_{ij}$  和  $A_{ij}$  就足够了。经验表明, 为每一个单独的值申请一个字节就已经足够。因此,  $n$  个物品所需要的总空间大小为  $n(n+1)$  字节。

提前计算矩阵  $\hat{A}$  的所有可能的项节省了构建  $\hat{A}$  不确定项所需要的冗长时间。快速检索  $\hat{A}$  中的相关项后, 我们可以通过解决式 (5.28) 所示的  $k \times k$  方程组系统得到插值权重。然而, 通过一个二次规划 [2] 把  $w$  限制为非负值时, 预测准确度可以得到适度提高。解决这个方程组是在 5.4.2 节描述的基于邻域的基本方法基础上增加的额外花费。对于  $k$  的典型值 (20~50), 额外的时间花费与计算  $k$  个最近的近邻所需要的时间相当。因此, 尽管与之前的方法相比, 该方法依赖于对插值权重进行更加详细的计算, 但是并没有显著增加运行时间, 参见文献 [2]。

#### 5.4.4 总结

基于邻域的插值的协同过滤算法或许是创建一个推荐系统最流行的方式。三个主要的组成部分描述了这个基于邻域的算法的特征: 1) 数据规范化; 2) 近邻的选择; 3) 插值权重的决定。

规范化对于一般意义上的协同过滤算法是至关重要的, 尤其是对于相对局部性的基于邻域的方法。否则, 即使再复杂的方法也注定会失败, 因为它们混合了不兼容的评分, 而这些评分是与不同规范化的用户或者物品相关的。我们基于基准预测器描述一种合适的方法来进行数据规范化。

邻域的选择是另一个重要的组件。它与采用的相似度度量直接相关。这里强调收缩不可靠相似度的重要性, 目的是避免发现有低评分支持度的近邻。

最后, 基于邻域的方法的成功依赖插值权重的选择, 这些插值权重被用来从已知评分的物品的近邻物品中估计未知的评分。然而, 大多数已知的方法缺少一种推导出这些权重的严密的方法。我们证明了如何把这些权重转化为最优化问题的全局解来求解, 这个最优化问题恰恰反映了它们的角色。

### 5.5 增强的基于邻域模型

大多数基于邻域的模型在本质上是局部性的, 因为它们只关注相关评分记录的一个小的子集。这与矩阵分解技术是相反的, 矩阵分解技术尽量从全局的角度来描述物品和用户的特征。看起来使用这种全局的观点可以提高准确性, 这正是提出本节中方法的动机。我们提出一个新的基于邻域的模型, 该模型借鉴了传统基于邻域的方法和矩阵分解模型两方面的原理。与其他基于邻域的模型一样, 该模型的基础也是基于物品的关系 (作为另一种选择, 或基于用户的关系), 这些关系为系统提供了一些之前讨论过的实际优势。同时, 与矩阵分解技术相似, 该模型是以一个全局优化框架为中心的, 这个框架通过考虑数据中存在的许多弱信号来提高准确度。

在 5.5.1 节中描述的主要方法允许我们用隐式反馈数据来丰富这个模型。除此之外, 这个主要方法提供了两种新的可能性。首先是在 5.5.2 节讨论的因子分解邻域模型, 该模型在计算效率上有很大的提高。其次是对时间效应的处理, 正如在 5.5.3 节所描述的那样, 这样做提高了预测准确度。

### 5.5.1 全局化的邻域模型

本节引入基于全局最优化的邻域模型。该模型提供了在 5.4.3 节中描述的模型的优点, 进而提供了更高的预测准确度。除此之外, 该模型还具有额外的优点, 概括如下:

- 不依赖任意的或者启发式的基于物品的相似度。这个新的模型饰演的是一个全局最优化问题的解。
- 固有的防止过拟合和“风险控制”的能力: 该模型恢复成了具有健壮性的基准预测器, 除非一个用户输入了足够多的相关评分。
- 该模型可以捕获包含在某个用户所有评分记录中弱信号的总量, 而不需要只关注最相似物品的几条评分记录。
- 该模型天生允许整合不同形式的用户输入, 如显式反馈和隐式反馈。
- 可以以线性时间复杂度和空间复杂度实现该模型的高度可扩展性, 因此使得基于物品的和基于用户的两种实现在非常大的数据集上有很好的可扩展性。
- 数据随时间变化的特点能够被整合到该模型中, 这样就提高了预测准确性, 参见 5.5.3 节。

#### 5.5.1.1 建立模型

我们通过不断修正公式来逐步构建模型的各个组成部分。之前的模型是以特定用户的插值权重为中心的, 也就是式(5.19)中的  $\theta_{ij}^u$  或者式(5.17)中的  $s_{ij} / \sum_{j \in S^k(i, u)} s_{ij}$ 。为了方便实现全局最优化, 我们放弃使用这样特定用户的权重, 而是使用与特定用户无关的基于物品的权重。物品  $j$  到物品  $i$  的权重定义为  $w_{ij}$ , 该权重从数据中通过最优化学习得到。模型的初始框架通过下面的等式来描述评分  $r_{ui}$ :

$$\hat{r}_{ui} = b_{ui} + \sum_{j \in R(u)} (r_{uj} - b_{uj}) w_{ij} \quad (5.29)$$

这个规则以原始但是具有健壮性的基准预测器( $b_{ui}$ )开始。然后, 估计值通过对用户  $u$  的所有评分求和来调整。

现在考虑插值权重。通常基于邻域的模型中的权重代表了插值系数, 这些系数把未知的评分和已知的评分联系起来。这里采用一个不同的观点, 该观点允许使用更灵活的权重。我们不再把权重当作插值系数。相反, 我们把权重当作调整或者补偿的一部分, 并把它们增加到基准预测器中。这样权重  $w_{ij}$  代表了我们基于观察到的  $r_{uj}$  值来提高  $r_{ui}$  的预测值的程度。对于两个相关的物品  $i$  和  $j$ , 我们希望  $w_{ij}$  的值较大。因此, 任何时候如果一个用户对物品  $j$  的评分超过预期值( $r_{uj} - b_{uj}$  很高), 我们将通过把  $(r_{uj} - b_{uj}) w_{ij}$  增加到基准预测来增加用户对物品  $i$  的估计值。相类似, 通过用户  $u$  正好按照期望( $r_{uj} - b_{uj}$  接近于 0)评分的物品  $j$ , 或者通过对物品  $i$  没有预测价值( $w_{ij}$  接近于 0)的物品  $j$ , 我们的估计值不会与基准值偏离得太多。

这种观点提出了对式(5.29)的几种增强。首先, 我们可以采用二元的用户输入形式, 这种形式对矩阵分解模型很有益。也就是说, 我们分析哪些物品被评分, 而不关注具体的评分值。为了这个目的, 我们增加另一组权重, 重写式(5.29), 如下所示:

$$\hat{r}_{ui} = b_{ui} + \sum_{j \in R(u)} [(r_{uj} - b_{uj}) w_{ij} + c_{ij}] \quad (5.30)$$

相类似, 这里我们可以使用另一组隐式反馈  $N(n)$ , 例如, 用户租借或购买的物品的集合, 得到下面的预测规则:



$$\hat{r}_{ui} = b_{ui} + \sum_{j \in R(u)} (r_{uj} - b_{uj})w_{ij} + \sum_{j \in N(u)} c_{ij} \quad (5.31)$$

与  $w_{ij}$  很相似,  $c_{ij}$  也是增加到基准预测器的补偿量。对于两个物品  $i$  和  $j$ , 用户  $u$  对物品  $j$  的隐式偏好使我们用  $c_{ij}$  来调整对  $r_{ui}$  的估计值。当由物品  $i$  可以预测物品  $j$  时,  $c_{ij}$  应该很大。

使用全局权重, 而不是特定用户的插值系数, 是为了强调缺失值的影响。也就是说, 一个用户的观点不仅体现在他评分的物品, 而且体现在他没有评分的物品。例如, 假设一个电影评分数据集显示, 对“怪物史莱克 3”评分高的用户也会对“怪物史莱克 1~2”评分较高。这将会建立从“怪物史莱克 1~2”到“怪物史莱克 3”的高权重。现在, 如果一个用户根本就没对“怪物史莱克 1~2”评分, 那他对“怪物史莱克 3”的评分将会被惩罚, 因为一些必需的权重没能增加到总和。

对于之前的模型式(5.17)和式(5.19), 由于它们从集合  $\{r_{uj} - b_{uj} | j \in S^k(i; u)\}$  中取值  $r_{ui} - b_{ui}$  来进行插值, 所以保持  $b_{ui}$  值和  $b_{uj}$  值的兼容性就很有必要了。然而, 这里不使用插值, 因此可以把  $b_{ui}$  和  $b_{uj}$  的定义解耦。相应地, 一个更加一般化的预测规则是  $\hat{r}_{ui} = b_{ui} + \sum_{j \in R(u)} (r_{uj} - b_{uj})w_{ij} + c_{ij}$ 。常量  $\hat{b}_{ui}$  代表了其他方法对  $r_{ui}$  进行的预测, 如隐语义模型。这里建议使用下面的预测规则, 该规则工作得很好:

$$\hat{r}_{ui} = \mu + b_u + b_i + \sum_{j \in R(u)} [(r_{uj} - b_{uj})w_{ij} + c_{ij}] \quad (5.32)$$

重要的是,  $b_{uj}$  是常量, 其推导过程如 5.2.1 节中解释的那样。然而,  $b_u$  和  $b_i$  变成了像  $w_{ij}$  和  $c_{ij}$  那样来优化的参数。

我们已经发现规范化模型中的和是很有益的, 于是得到下面的形式:

$$\hat{r}_{ui} = \mu + b_u + b_i + |R(u)|^{-\alpha} \sum_{j \in R(u)} [(r_{uj} - b_{uj})w_{ij} + c_{ij}] \quad (5.33)$$

常量  $\alpha$  控制了规范化的程度。一种非规范化的规则 ( $\alpha=0$ ) 会使提供很多评分记录 ( $|R(u)|$  很高) 的用户的预测值跟基准预测有较大的偏置。另一方面, 一个完全规范化的规则消除了评分记录的个数在预测值与基准预测的偏置方面的影响。在许多情况下, 对于评分记录很多的用户, 让他们的预测值与基准预测的偏置较大对于推荐系统来说是一个很好的尝试。这种情形下, 对那些与模型吻合得很好且提供了很多输入的用户进行预测时, 就会有更大的风险。对于这样的用户, 我们宁愿给出诡异且不常见的推荐。同时, 我们不太确定如何对那些只提供很少输入的用户建模, 这种情形下, 我们将会使用接近基准值的保险估计值。在 Netflix 数据集上的经验表明, 当  $\alpha=0.5$  时, 模型取得最好的结果, 也就是下面的预测规则:

$$\hat{r}_{ui} = \mu + b_u + b_i + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} [(r_{uj} - b_{uj})w_{ij} + c_{ij}] \quad (5.34)$$

作为一种可选的改进, 可以通过剪掉那些与不太可能的基于物品的关联相对应的参数来降低模型的复杂度。把  $S^k(i)$  定义为与物品  $i$  最相似的  $k$  个物品的集合, 该集合是由如相似度量  $s_{ij}$  或与物品集相关联的自然层次决定的。除此之外, 定义  $R^k(i; u) \stackrel{\text{def}}{=} R(u) \cap S^k(i)^\ominus$ 。现在, 当根据规则(5.34)来预测  $r_{ui}$  时, 我们期望影响最大的权重能够跟与物品  $i$

⊖ 标记说明: 在其他邻域模型最好采用  $S^k(i; u)$ , 表示用户  $u$  评分的物品中, 与  $i$  最相似的  $k$  个物品。因此, 如果用  $u$  至少评分了  $k$  个物品, 我们将总会有  $|S^k(i; u)| = k$ , 不管那些物品与  $i$  有多相似。但是,  $|R^k(i; u)|$  通常都会比  $k$  小, 因为有些和  $i$  最相似的物品没有被用户  $u$  评分。

相似的物品关联起来。因此,用下面的预测规则来替代式(5.34):

$$\hat{r}_{ui} = \mu + b_u + b_i + |R^k(i; u)|^{-\frac{1}{2}} \sum_{j \in R^k(i; u)} [(r_{uj} - b_{uj})w_{ij} + c_{ij}] \quad (5.35)$$

当  $k=\infty$  时, 规则(5.34)与式(5.35)是一致的。然而, 对于  $k$  的其他值, 模型(5.35)提供了显著减少涉及的变量的可能性。

### 5.5.1.2 参数估计

预测规则(5.35)允许我们快速在线预测。在需要进行参数估计的预处理阶段, 我们需要做更多的计算工作。这个新的基于邻域的模型的一个主要设计目标是使高效的全局最优化过程成为可能, 而这正是之前基于邻域的模型所缺少的。因此, 模型的参数是通过解决与式(5.35)相关的正则化最小二乘法问题学习得到的:

$$\begin{aligned} \min_{b_u, w_u, c_u, (u, i) \in K} & \sum (r_{ui} - \mu - b_i - b_u - |R^k(i; u)|^{-\frac{1}{2}} \sum_{j \in R^k(i; u)} ((r_{uj} - b_{uj})w_{ij} + c_{ij}))^2 \\ & + \lambda_{10} (b_u^2 + b_i^2 + \sum_{j \in R^k(i; u)} w_{ij}^2 + c_{ij}^2) \end{aligned} \quad (5.36)$$

这个凸问题的最优解可以通过最小二乘法解算器得到, 最小二乘法解算器是标准线性代数包的一部分。然而, 我们发现通过下面的随机梯度下降算法可以更快地得到答案。我们定义准确性误差  $e_{ui}$ , 其中,  $e_{ui} = r_{ui} - \hat{r}_{ui}$ 。我们对集合  $k$  中所有已知的评分记录做循环。对于一个给定的训练案例  $r_{ui}$ , 我们通过朝着与梯度相反的方向移动来修正参数, 如下所示:

$$\begin{aligned} & \bullet \quad b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda_{10} \cdot b_u) \\ & \bullet \quad b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda_{10} \cdot b_i) \\ & \bullet \quad \forall j \in R^k(i; u): \\ & \quad w_{ij} \leftarrow w_{ij} + \gamma \cdot (|R^k(i; u)|^{-\frac{1}{2}} \cdot e_{ui} \cdot (r_{uj} - b_{uj}) - \lambda_{10} \cdot w_{ij}) \\ & \quad c_{ij} \leftarrow c_{ij} + \gamma \cdot (|R^k(i; u)|^{-\frac{1}{2}} \cdot e_{ui} - \lambda_{10} \cdot c_{ij}) \end{aligned}$$

元参数  $\gamma$ (步长)和  $\lambda_{10}$  由交叉验证决定。在 Netflix 数据中, 我们使用  $\gamma=0.005$ ,  $\lambda_{10}=0.002$ , 另一个重要参数是  $k$ , 该参数控制了邻域的大小。经验表明随着  $k$  的增大, 我们在测试集上得到的结果的准确性也随之提高。因此,  $k$  的选择反映了预测准确性和计算代价之间的一种权衡。在 5.5.2 节中, 我们将会介绍该模型的因子化版本, 该版本允许我们在  $k=\infty$  条件下工作, 因此我们能得到最准确的结果, 同时又降低了运行时间。

在训练数据上的循环次数的典型值是 15~20。至于每次迭代的时间复杂度, 我们分析的  $k=\infty$  最准确情形, 也就是使用预测规则(5.34)的情形。对于每一个用户  $u$  和物品  $i \in R(u)$ , 我们需要修改  $\{w_{ij}, c_{ij} | j \in R(u)\}$ 。因此训练阶段的总体时间复杂度为  $O(\sum_u |R(u)|^2)$ 。

### 5.5.1.3 准确性比较

图 5.1 展示了 Netflix 数据中使用全局最优化邻域模型(以后记为 GlobalNgbr)得到的实验结果。我们在参数  $k$  取不同值条件下研究了该模型。带正方形标记的实黑线显示准确性随  $k$  的增加单调增加, 由于根均方差(RMSE)从当  $k=250$  时的 0.9139 降低到当  $k=\infty$  的 0.9002。(注意 Netflix 数据包含 17 770 部电影,  $k=\infty$  等价于  $k=17 769$ 。)我们使用隐式反馈来重复进行该实验, 也就是说, 从模型中去掉  $c_{ij}$  参数。带 X 标记的实黑曲

线描述的结果显示：随着  $k$  增加，估计准确性会显著下降。这验证了把隐式反馈整合到模型的价值。

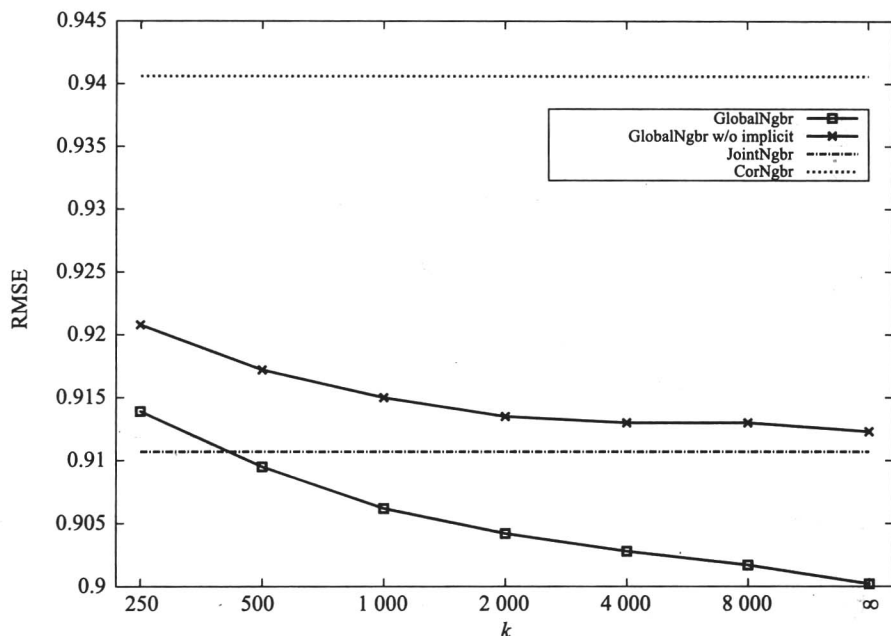


图 5.1 基于邻域的模型的比较。准确度是由 Netflix 测试集上的 RMSE 来度量的，因此 RMSE 越小，准确度越高。我们在考虑隐式反馈和不考虑隐式反馈两种条件下度量全局化的优化模型(GlobalNgr)的准确度。RMSE 表现为变量的函数，变量  $k$  代表邻域大小。另两种模型的准确度显示为两条平行线，我们为每一种模型选择一个最优的邻域大小

为了对比，我们提供两种之前描述的基于邻域的模型的结果。第一个是基于相似度的邻域模型(在 5.4.2 节中有描述)，该模型是文献中最流行的方法。我们把这种模型记为 CorNgr。第二个是在 5.4.3 节中描述的一个更加准确的模型，记为 JointNgr。对于这两种模型，我们试图选择最优的参数和邻域大小。对于模型 CorNgr，邻域大小取 20；对于模型 JointCorNgr，邻域大小取 50。这两种模型的结果分别由点线和虚线描述。显然流行的 CorNgr 方法的准确度比其他基于邻域的模型的准确度显著要低。相反，与 JointNgr 方法相比，GlobalNgr 方法的结果更加准确，只要  $k$  的值至少是 500。注意参数的值( $x$  坐标轴)是与之之前的模型是不相关的，因为这些模型的邻域的概念不同使得邻域大小是不兼容的。然而，我们观察到尽管 GlobalNgr 模型的性能随着更多近邻的加入而不断提高，但这对于另外两种模型并不适用。对于 CorNgr 模型和 JointNgr 模型，它们的性能在邻域的大小相对较小时取得最佳值，随后随着邻域的大小的增大而下降。这或许可以通过这样的一个事实来解释，即在 GlobalNgr 模型中，参数是通过一个正式的优化过程直接从数据中学习得到的，在该优化过程中，参数越多，推荐结果越有效。

最后，我们考虑运行时间。尽管之前基于邻域的模型需要非常细微的预处理，但是 JointNgr 模型[2]却需要为每一个预测解决一个小方程组问题。当估计参数时，这个新模型确实涉及预处理。然而，在线预测可以通过规则(5.35)立即得到。预处理的时间随着的增加而增大。图 5.2 显示该模型 Netflix 数据上迭代一次所需要的典型运行时间，该时间是在主频为 3.4GHz、CPU 为奔腾 4 的单处理器上测量得到的。

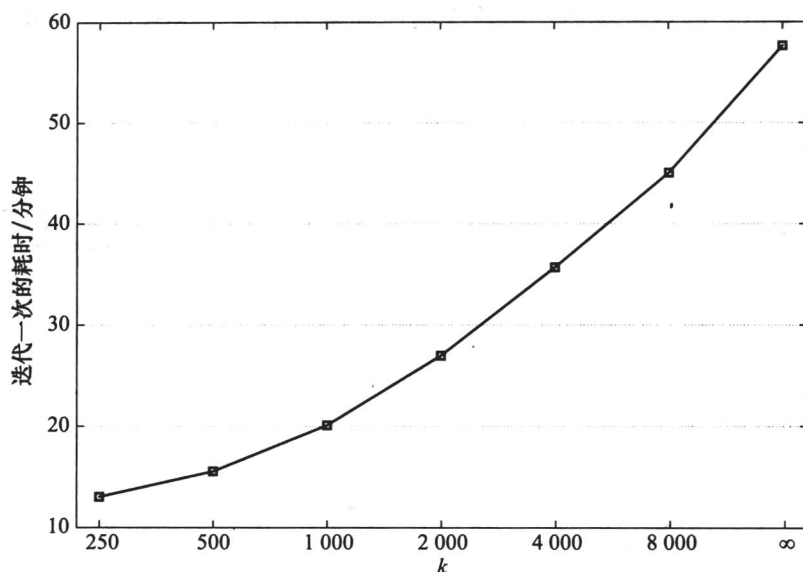


图 5.2 全局优化的基于邻域模型迭代一次的运行时间, 该运行时间是参数  $k$  的函数

## 5.5.2 因式分解的邻域模型

在 5.5.1 节中, 我们展示了一个更加准确的基于邻域模型, 该模型是基于预测规则 (5.34) 的, 并且其训练复杂度为  $O(\sum_u |R(u)|^2)$ , 空间复杂度为  $O(m+n^2)$ 。( $m$  是用户的个数,  $n$  是物品的个数。) 我们可以通过剪掉不可能的基于物品的关联来简化该模型, 进而改善时间和空间复杂度。简化程度是由参数  $k \ll n$  控制的, 模型的简化降低了运行时间而且允许空间复杂度为  $O(m+nk)$ 。然而, 随着  $k$  变小, 模型的准确性也随之下降。除此之外, 模型的简化需要依赖一个外部的非自然的相似度量, 而这种相似度量正是我们想避免的。因此, 我们将会展示如何在保留完全的致密规则的准确度的条件的同时, 显著降低时间和空间复杂度。

### 5.5.2.1 把基于物品的关系进行因式分解

我们通过把物品  $i$  与三个向量关联起来从而把基于物品的关系包含到模型中:  $q_i, x_i, y_i \in \mathbf{R}^f$ 。通过这种方式, 把  $w_{ij}$  限制为  $q_i^T x_j$ 。相类似, 强制  $c_{ij} = q_i^T y_j$ 。从根本上讲, 这些向量试图把物品映射到一个  $f$  维隐语义空间中。在这个隐语义空间中, 这些向量衡量了数据的各方面, 这些方面是通过从数据中学习而自动揭示的。通过把上面的等式代入模型 (5.34), 我们得到下面的预测规则:

$$\hat{r}_{ui} = \mu + b_u + b_i + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} [(r_{uj} - b_{uj}) q_i^T x_j + q_i^T y_j] \quad (5.37)$$

通过下面等价的规则可以明显提高计算效率:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T (|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} (r_{uj} - b_{uj}) x_j + y_j) \quad (5.38)$$

注意, 预测规则 (5.38) 的大部分 ( $|R(u)|^{-\frac{1}{2}} \sum (r_{uj} - b_{uj}) x_j + y_j$ ) 只依赖于用户  $u$ , 而与物品  $i$  无关。这样就可以通过一个高效的方式来学习模型的参数。与往常一样, 最小化以下与式 (5.38) 相关的平方误差函数:

$$\min_{q_i, x_i, y_i, b_u, b_i, (u,i) \in K} \sum (r_{ui} - \mu - b_u - b_i - q_i^T (|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} (r_{uj} - b_{uj}) x_j + y_j))^2 + \lambda_{11} (b_u^2 + b_i^2 + \|q_i\|^2 + \sum_{j \in R(u)} \|x_j\|^2 + \|y_j\|^2) \quad (5.39)$$

最优化过程是通过随机梯度下降算法完成的，随机梯度下降算法可以用下面的伪代码描述：

```

LearnFactorizedNeighborhoodModel(Known ratings:  $r_{ui}$ , rank:  $f$ )
% For each item  $i$  compute  $q_i, x_i, y_i \in \mathbf{R}^f$ 
% which form a neighborhood model
Const #Iterations = 20,  $\gamma = 0.002, \lambda = 0.04$ 
% Gradient descent sweeps:
for count = 1, ..., #Iterations do
    for  $u = 1, \dots, m$  do
        % Compute the component independent of  $i$ :
         $p_u \leftarrow |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} (r_{uj} - b_{uj}) x_j + y_j$ 
         $sum \leftarrow 0$ 
        for all  $i \in R(u)$  do
             $\hat{r}_{ui} \leftarrow \mu + b_u + b_i + q_i^T p_u$ 
             $e_{ui} \leftarrow r_{ui} - \hat{r}_{ui}$ 
            % Accumulate information for gradient steps on  $x_i, y_i$ :
             $sum \leftarrow sum + e_{ui} \cdot q_i$ 
            % Perform gradient step on  $q_i, b_u, b_i$ :
             $q_i \leftarrow q_i + \gamma \cdot (e_{ui} \cdot p_u - \lambda \cdot q_i)$ 
             $b_u \leftarrow b_u + \gamma \cdot (e_{ui} - \lambda \cdot b_u)$ 
             $b_i \leftarrow b_i + \gamma \cdot (e_{ui} - \lambda \cdot b_i)$ 
        for all  $i \in R(u)$  do
            % Perform gradient step on  $x_i$ :
             $x_i \leftarrow x_i + \gamma \cdot (|R(u)|^{-\frac{1}{2}} \cdot (r_{ui} - b_{ui}) \cdot sum - \lambda \cdot x_i)$ 
            % Perform gradient step on  $y_i$ :
             $y_i \leftarrow y_i + \gamma \cdot (|R(u)|^{-\frac{1}{2}} \cdot sum - \lambda \cdot y_i)$ 
return  $\{q_i, x_i, y_i | i = 1, \dots, n\}$ 

```

该模型的时间复杂度与输入呈线性关系，是  $O(f \cdot \sum_u |R(u)|)$ 。该模型的时间复杂度显著优于非因子分解的模型，后者的时间复杂度为  $O(\sum_u |R(u)|^2)$ 。我们在Netflix数据上度量该模型的性能，参见表 5.3。我们可以使用更多的因子（增大  $f$ ）来提高准确度。然而，如果因子的个数超过 200，准确度几乎不会再提高，却降低了运行时间。有趣的是，如果该模型的因子个数超过 200 ( $f \geq 200$ )，与非因子分解的模型 ( $k = \infty$ ) 相比，该模型提高的准确度几乎可以忽略。此外，增加的时间复杂度通过在运行时间上的区别体现出来。例如，因子个数  $f = 200$  的非因子分解模型 ( $k = \infty$ ) 的一次迭代的时间大概是 58 分钟。另一方面，因子个数为 200 的因子分解模型迭代一次的时间为 14 分钟，同时几乎不降低准确度。

表 5.3 因子分解的基于物品的邻域模型的性能。因子个数超过 200 时，该模型的性能略优于非因子分解模型，非因子分解模型的运行时间相对更短

因子	50	100	200	500
EMSE	0.903 7	0.901 3	0.900 0	0.899 8
时间/迭代	4.5min	8min	14min	34min

因子分解模型的最有利的地方在于它降低了空间复杂度，该模型的空间复杂度为  $O(m + nf)$ ，与输入呈线性关系。之前基于邻域的模型需要存储物品之间的所有成对关系，



因此其空间复杂度为  $O(m+n^2)$ 。例如, 对于包含了 17 770 部电影的 Netflix 数据集, 这样的平方数量级空间仍能够放到主存中。一些商用推荐系统会处理更多的物品。例如, 像 Netflix 这样提供在线租赁业务的服务目前拥有超过 100 000 个主题。音乐下载商铺甚至会有更多的主题。这样拥有 100 000 个物品的更加综合性的系统最终会需要借助外排序来处理所有成对关系的集合。然而, 随着物品的数量增长到百万级别[20], 如 Amazon 的基于物品的推荐系统, 设计者必须保留成对关系的一个简化版本。为了达到这个目的, 只存储某件物品的 top- $k$  近邻的评分, 这样就把空间复杂度降低到  $O(m+nk)$ 。然而, 这样的简化技术将不可避免地降低准确度, 原因是该技术缺失了重要的关系, 这一点在之前的章节中就说明过。除此之外, 在一个高维空间中找到 top- $k$  最近邻并非易事, 它需要相当大的计算代价。所有的这些问题在因子分解邻域模型中都没有出现, 该模型在不损失准确度的条件下提供了一个线性的时间和空间复杂度。

因子分解邻域模型与某些隐语义模型很像。这里重要的区别是我们是对物品—物品关系进行了因式分解, 而不是评分本身。表 5.3 报告的结果与广泛使用的 SVD 模型旗鼓相当, 但却不如 SVD++ 模型的结果, 参见 5.3 节。然而, 该因子分解的邻域模型保留了之前讨论的传统邻域模型的实用优点, 即解释推荐结果和立即反映新评分的能力。

作为一个辅助的备注, 我们想说明使用三个独立的因子集合的目的是增加灵活性。确实, 在 Netflix 数据集中, 这样做让我们得到了最准确的结果。然而, 另一种合理的选择是使用小规模向量集合, 例如, 可以令  $q_i = x_i$  (表示权重是对称的:  $w_{ij} = w_{ji}$ )。

### 5.5.2.2 基于用户的模型

基于用户的邻域模型通过考虑志趣相同的人如何对某件物品的评分来预测该物品的评分。这样的模型可以通过在基于物品的模型的推导过程中变换用户和物品的角色来实现。这里集中讨论基于物品的模型, 该模型是式(5.34)所示的基于物品的模型的对偶形式。主要的区别是我们用关联用户对的权重来代替关联物品对的权重  $w_{ij}$ :

$$\hat{r}_{ui} = \mu + b_u + b_i + |R(i)|^{-\frac{1}{2}} \sum_{v \in R(i)} (r_{vi} - b_{vi}) w_{uv} \quad (5.40)$$

集合  $R(i)$  包含了对物品  $i$  评分的所有用户。注意这里决定不考虑隐式反馈。这是因为增加这样的反馈对于工作在 Netflix 数据上的基于用户的模型不是很有利。

基于用户的模型在多种场景中都很有用。例如, 一些推荐系统或许会处理会被迅速替换的物品, 因此基于物品的关联变得很不稳定。另一方面, 一个稳定的用户群会建立物品之间的长期的关系。这种场景的一个例子是对 Web 文章和新物品的推荐系统, 这些 Web 文章和新物品在本质上是不断变化的, 参见文献[8]。在这些场景中, 以基于用户的关系为中心的系统更具吸引力。

除此之外, 基于用户的方法能够识别出不同种类的、基于物品的方法不能识别的关系, 因此该方法在特定场合中很有用。例如, 假设想要预测  $r_{ui}$ , 但是用户  $u$  评分的物品没有一个是真正与物品  $i$  相关的。这种情形下, 基于物品的方法将会遇到明显的困难。然而, 当使用一个基于用户的观点时, 我们也许能找到与对物品  $i$  评分的用户  $u$  相似的用户集合。这些用户对物品  $i$  的评分将有助于提高  $r_{ui}$  的预测准确度。

基于用户的模型的最大缺点体现在计算效率上。由于一般情况下用户的数量比物品的数量多, 所以提前计算并存储所有的基于用户的关系, 甚至是一个合理的简化版本, 都需要很昂贵的代价甚至是完全不切合实际的。除了高达  $O(m^2)$  的空间复杂度, 优化模型式(5.40)的时间复杂度也比基于物品的模型的时间复杂度要高, 达到了  $O(\sum_i |R(i)|^2)$  (注

意  $|R(i)|$  一般情况下比  $|R(u)|$  大)。这些问题使得基于用户的模型在实际中并不实用。

**因子分解模型。**当沿着与基于物品的模型相同的路线因子分解基于用户的模型时，所有这些计算式的区别都不存在了。现在，我们把每一个用户与两个向量  $p_u$ 、 $z_u$  关联起来。假设基于用户的关系结构化为  $w_{uv} = p_u^T z_v$ 。把上面的等式代入式(5.40)中，得

$$\hat{r}_{ui} = \mu + b_u + b_i + |R(i)|^{-\frac{1}{2}} \sum_{v \in R(i)} (r_{vi} - b_{vi}) p_u^T z_v \quad (5.41)$$

再一次，我们通过在独立的总值中引入一个依赖于物品  $i$  而与用户  $u$  无关的项来提高计算效率。因此预测规则如下所示：

$$\hat{r}_{ui} = \mu + b_u + b_i + p_u^T |R(i)|^{-\frac{1}{2}} \sum_{v \in R(i)} (r_{vi} - b_{vi}) z_v \quad (5.42)$$

在基于物品的模型的一个并行方式中，所有参数都可以在线性时间  $O(f \cdot \sum_i |R(i)|)$  内学习得到。空间复杂度也与输入呈线性增长，为  $O(n + mf)$ 。与之前已知的结果相比，该并行方式显著降低了基于用户的模型的复杂度，参见表 5.4。我们应该说明，与基于物品的模型不同，在实现基于用户的模型时没有考虑隐式反馈，这样或许会降低运行时间。基于用户的模型的准确度显著优于广泛使用的基于关系的物品间模型的准确度，正如图 5.1 报告的那样，该模型的 RMSE=0.9406。此外，准确度略优于基于物品的模型的变种，这个变种没有考虑隐式反馈(图 5.1 中的黄线所示)。考虑到基于物品的方法比基于用户的方法更加准确，这个结果是令人吃惊的。看起来如果一个基于用户的模型实现得很好，其速度和准确度能够比得上基于物品的模型。然而，如果考虑隐式反馈，基于物品的模型的性能可以显著提高。

表 5.4 因式分解的邻域模型的性能

因子	50	100	200	500
EMSE	0.911 9	0.911 0	0.910 1	0.909 3
时间/迭代	3min	5min	8.5min	18min

**混合基于物品的模型和基于用户的模型。**由于基于物品的模型和基于用户的模型解决了数据的不同方面的问题，总体准确度可以通过组合两种模型的预测结果来提高。这样的方法之前就提到过，并且证明可以提高准确度，参见文献[4, 32]。然而，过去的工作是基于在后处理阶段混合基于物品的模型和基于用户的模型的预测结果，而每个单独的模型在训练时是独立于其他模型的。一种更合理的方法是同时训练两种单独的模型，使它们在学习参数时就能相互了解。因此，在整个训练阶段每个模型都能知道另一种模型的性能并尽力来补充。本节的方法认为基于邻域的模型等价于一个形式化的优化问题，因此可以自然地处理。我们设计一个模型，该模型对基于物品的模型(5.37)和基于用户的模型(5.41)求和，如下所示：

$$\begin{aligned} \hat{r}_{ui} = & \mu + b_u + b_i + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} [(r_{uj} - b_{uj}) q_i^T x_j + q_i^T y_j] \\ & + |R(i)|^{-\frac{1}{2}} \sum_{v \in R(i)} (r_{vi} - b_{vi}) p_u^T z_v \end{aligned} \quad (5.43)$$

模型参数通过对相关的平方误差函数进行一个随机化梯度下降来学习得到。Netflix 数据上的实验显示预测准确度确实比每一个单独的模型要好。例如，当因子个数为 100 时，该模型的 RMSE 为 0.8996；当因子个数为 200 时，得到的 RMSE 为 0.8953。

这里，需要说明，我们的方法可以以同样的方式把基于邻域的模型和完全不同的模型

集成起来。例如,在文献[17]中,我们展示了一种集成的模型,该模型把基于物品的模型和隐语义模型(SVD++)结合起来,因此提高了预测准确度,它把 RMSE 降低到 0.887 以下。因此,在考虑整合基于物品的模型和基于用户的模型时,应该考虑具有潜在的更好准确度的可能性。

### 5.5.3 基于邻域的模型的动态时序

基于全局优化(5.5.1 节中描述的)的基于物品的模型的优点之一就是它使我们能够以一种合理的方式捕获时间效应。正如我们之前说明的那样,用户偏好随着时间而变化,因此在 CF 模型中引入时序方面就很重要了。

当应用规则(5.34)来处理时间效应时,两个组成部分应该分开考虑。第一部分,  $\mu + b_i + b_u$ , 与基准预测部分相对应。典型意义上,这个部分解释了观察到的信息中的变化性。第二个部分,  $|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} (r_{uj} - b_{uj})w_{ij} + c_{ij}$  捕获了信息量更大的信息,因为该信息处理了用户-物品交互信息。至于基准部分,由于与因子模型相比并没有变化,所以根据式(5.6)和式(5.9)可以用  $\mu + b_i(t_{ui}) + b_u(t_{ui})$  代替。然而,捕获交互部分内时间效应需要一个不同的策略。

基于物品的权重( $w_{ij}$  和  $c_{ij}$ )反映了物品的固有特点,因此不会随时间变化。学习过程应该捕获无偏的长期值,而不应该过多地受随时间变化的方面的影响。实际上,如果处理不当,数据随时间变化的本性将掩盖大部分长期的基于物品的关系。例如,一个用户在一个时间周期内同时对物品  $i$  和  $j$  给予了很高的评分,这将是这两个物品有关联的一个很好的指示,因此  $w_{ij}$  的值就较高。另一方面,尽管用户的兴趣(如果其身份不变)会随着时间改变,如果那两个评分给出的时间相隔了 5 年,这并不能表明这两个物品间有关系。此外,我们认为这些考虑几乎是依赖用户的;一些用户的兴趣比其他用户更加一致,并允许把他们的长期行为关联起来。

我们的目标是为基于物品的权重提取准确值,尽管有时间效应的影响。首先我们需要把用户  $u$  评分的两个物品之间的不断衰弱的关联参数化。我们采用函数  $e^{-\beta_u \cdot \Delta t}$  形式的指数衰减,其中  $\beta_u > 0$ , 该参数控制了特定用户的衰减速度并从数据中学习得到。我们也以其他衰减形式进行试验,比如,更加容易计算的  $(1 + \beta_u \Delta t)^{-1}$ , 使用该衰减形式的结果具有相同的准确度,同时又降低了运行时间。

这样就可以得到下面的预测规则:

$$\hat{r}_{ui} = \mu + b_i(t_{ui}) + b_u(t_{ui}) + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} e^{-\beta_u \cdot |t_{ui} - t_{uj}|} ((r_{uj} - b_{uj})w_{ij} + c_{ij}) \quad (5.44)$$

涉及的参数,  $b_i(t_{ui}) = b_i + b_{i, \text{Bin}(t_{ui})}$ ,  $b_u(t_{ui}) = b_u + \alpha_u \cdot \text{dev}_u(t_{ui}) + b_{u, t_{ui}}$ ,  $\beta_u$ ,  $w_{ij}$  和  $c_{ij}$  可以通过最小化相关的正则化平方误差学习得到:

$$\begin{aligned} & \sum_{(u,i) \in K} (r_{ui} - \mu - b_i - b_{i, \text{Bin}(t_{ui})} - b_u - \alpha_u \cdot \text{dev}_u(t_{ui}) - b_{u, t_{ui}} \\ & - |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} e^{-\beta_u \cdot |t_{ui} - t_{uj}|} ((r_{uj} - b_{uj})w_{ij} + c_{ij}))^2 \\ & + \lambda_{12} (b_i^2 + b_{i, \text{Bin}(t_{ui})}^2 + b_u^2 + \alpha_u^2 + b_{u, t_{ui}}^2 + w_{ij}^2 + c_{ij}^2) \end{aligned} \quad (5.45)$$

最小化过程是通过随机梯度下降完成的。取  $\lambda_{12} = 0.002$ , 步长(学习速率)取 0.005, 并迭代该过程 25 次。一个例外的情况是在更新  $\beta_u$  时,我们使用一个很小的步长  $10^{-7}$ 。训练时间复杂度与原始算法一样,为  $O(\sum_u |R(u)|^2)$ 。我们可以通过简化 5.5.1 节所解释

的基于物品的关联的集合来在复杂度和准确度上做出权衡。

与因子模型相同,在基于邻域的模型中,适当地考虑时间效应提高了在电影评分数据中的准确度。RMSE 从 0.9002[17]降低到 0.8885。据称,这显著优于之前已知的基于邻域的方法提供的结果。从某种角度看,这个结果甚至比使用混合方法(比如,把一个基于邻域的方法应用到方法[2, 23, 31]的残差上面)报告的结果更准确。我们得到的一个经验是:与设计更复杂的学习算法相比,考虑数据中的时间效应能够对预测准确度有更大的影响。

我们想强调一个有趣的观点。假设  $u$  是一个用户,其偏好变化得很快( $\beta_u$  很大)。因此,用户  $u$  之前的评分不应该对  $u$  在当前时刻  $t$  的状态有很大的影响。我们可以衰减用户  $u$  之前评分的权重,通过下面的代价函数得到“实例加权”:

$$\sum_{(u,i) \in \mathcal{K}} e^{-\beta_u \cdot |t-t_{ui}|} (r_{ui} - \mu - b_i - b_{i, \text{Bin}(t_{ui})} - b_u - \alpha_u \text{dev}_u(t_{ui}) - b_{u, t_{ui}} - |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} ((r_{uj} - b_{uj})w_{ij} + c_{ij}))^2 + \lambda_{12}(\dots)$$

这样的函数集中在用户在时刻  $t$  的现状,而不在强调用户之前的行为。我们或许会反对这个选择,而是选择为所有之前的、如式(5.45)所示的评分的预测误差赋予同等的权重,这样就对之前用户所有的行为进行了建模。因此,同等权重使得我们可以使用用户之前的每一个评分的信息,该信息被提炼成基于物品的权重。学习这些权重将会同等受益于某用户的所有评分。也就是说,如果用户在短时期内同时对两个物品给予了相似的评分,即使这是很久之前发生的,我们仍然可以推断出这两个物品是有关联的。

#### 5.5.4 总结

本节讲述了一个非传统的基于邻域的模型。不像之前的基于邻域的模型,该模型是基于形式化的优化一个全局代价函数。得到的模型不再是局部化的了,该模型考虑了一个由强关联的近邻组成的小集合之间的关联,而没有考虑所有成对关系。这样可以提高准确度,同时又保留了基于邻域的模型的一些优点,比如,预测结果的可解释性和在不用重新训练模型的情况下处理新评分(或新用户)的能力。

形式化的优化框架提供了几个新的可能性。第一个是基于邻域的模型的因子分解版本,该版本改善了计算复杂度,同时保留预测准确性。尤其是,限制之前基于邻域的模型的平方级别的存储要求在该模型中也消失了。

第二个是把时间效应整合到该模型中。为了揭示物品之间的准确关联,我们提出了一个模型,该模型能够学习某一用户评分的两个物品之间的影响是如何随时间衰减的。与矩阵分解模型很类似,考虑时间效应可以显著提高预测准确度。

### 5.6 基于邻域的模型和因子分解模型的比较

本节是围绕协同过滤领域两种不同的方法组织的:因子分解方法和基于邻域的方法。每一种方法都是从不同的基本原理演进过来的,因此得到了不同的预测规则。我们也认为因子分解模型能够得到更准确的结果,而基于邻域的模型有一些实用的优点。本节将会证明尽管有这些区别,这两种方法还是有很多相似的部分。毕竟,它们都是线性模型。

考虑 5.3.1 节中的 SVD 模型。该模型是基于下面等式的:

$$\hat{r}_{ui} = q_i^T p_u \quad (5.46)$$

为了渐变, 在这里忽略了基准预测器, 但是我们可以很容易地把它们重新引入该模型, 或仅假设它们在早期阶段从所有评分中减掉了。

我们把所有的物品因子放到  $n \times f$  矩阵  $Q$  中, 其中  $Q = [q_1, q_2, \dots, q_n]^T$ 。相似地, 我们把所有的用户因子放到  $n \times f$  矩阵  $P$  中, 其中  $P = [p_1, p_2, \dots, p_m]^T$ 。我们用  $n_u \times f$  矩阵  $Q[u]$  来定义矩阵  $Q$  中与用户  $u$  评分相关的那部分, 其中  $n_u = |R(u)|$ 。假设向量  $r_u$  包含了用户  $u$  的评分, 该评分顺序和矩阵  $Q[u]$  中的评分顺序一致。现在, 对用户  $u$  的所有评分使用式(5.46), 我们可以以一个矩阵的形式重新定义该公式:

$$\hat{r}_u = Q[u]p_u \quad (5.47)$$

给定  $Q[u]$ 、 $\|r_u - Q[u]p_u\|_2$ , 在下面条件下取得最小值:

$$p_u = (Q[u]^T Q[u])^{-1} Q[u]^T r_u$$

实际上, 我们可以用一个非负的参数  $\lambda$  来正则化上面的等式:

$$p_u = (Q[u]^T Q[u] + \lambda I)^{-1} Q[u]^T r_u$$

把上面的  $p_u$  代入式(5.47), 得

$$\hat{r}_u = Q[u](Q[u]^T Q[u] + \lambda I)^{-1} Q[u]^T r_u \quad (5.48)$$

可以引入一些新的符号来简化这个表达式。我们把  $f \times f$  矩阵  $(Q[u]^T Q[u] + \lambda I)^{-1}$  定义为  $W^u$ , 该矩阵是与用户  $u$  相关联的权重矩阵。相应地, 从用户  $u$  的角度看, 物品  $i$  和  $j$  之间加权的相似度定义为  $s_{ij}^u = q_i^T W^u q_j$ 。使用这种符号以及式(5.48), 用户  $u$  对物品  $i$  的预测偏好在 SVD 模型上的估计值如下:

$$\hat{r}_{ui} = \sum_{j \in R(u)} s_{ij}^u r_{uj} \quad (5.49)$$

我们可以把 SVD 模型简化为一个线性模型, 该线性模型把用户的偏好预测为过去行为的函数, 并由基于物品的相似度加权。每一个过去的行为都会有一个独立的项, 这些项共同形成了预测值  $\hat{r}_{ui}$ 。这与基于邻域的基于物品的模型是等价的。令人惊讶的是, 我们把矩阵因子分解模型转换成一个基于物品的模型, 该模型具有下面的特征:

- 插值是从用户过去的所有评分中得到的, 而不仅是与跟当前物品最相似的物品相关的评分。
- 关联物品  $i$  和  $j$  的权重被分解为两个向量的内积, 一个向量与用户  $i$  相关, 另一个向量与物品  $j$  相关。
- 基于物品的权重与特定用户的规范化有关, 这是通过矩阵  $W^u$  实现的。

这些特性支持了我们在如何最好地构造一个基于邻域的模型方面的研究成果。首先, 我们在 5.5.1 节中证明了当邻域(由参数控制)取最大值时, 基于邻域的模型取得最好的结果。因为当邻域大小取最大值时, 过去用户所有的评分都被考虑到了。其次, 在 5.5.2 节中, 我们尝试了因子分解物品-物品权重矩阵。至于特定用户的规范化, 我们使用一个简单的规范化器:  $n_u^{-0.5}$ 。SVD 模型可能会使用  $W^u$  来进行更加基础的规范化, 这种规范化工作得很好。然而, 实际中计算  $W^u$  的代价很昂贵。我们提出的基于物品的模型和由 SVD 模型隐式推导出的模型间的另一个区别是, 我们在基于用户的模型中采用非对称的权重 ( $w_{ij} \neq w_{ji}$ ), 而 SVD 模型推导的模型中  $s_{ij}^u = s_{ji}^u$ 。

在上面的推导过程中, 我们展示了如何由 SVD 模型推导出等价的基于物品的技术。通过把  $q_2$  作为评分和用户因子的函数, 我们可以以一种完全类似的方法推导出等价的基于用户的技术。这样可以得到三种等价的模型: SVD、基于物品的模型和基于用户的模型。除了可以把 SVD 模型和基于邻域的模型链接在一起, 可以证明, 只要设计得好, 基



于用户的方法和基于物品的方法是等价的。

最后一组关系(基于用户和基于物品之间的关系)也可以直观地得到。基于邻域的模型试图通过遵循用户—物品邻接链而把用户和新物品关联起来。这样的邻接代表了各自用户和物品之间的偏好或者评分关系。基于用户的和基于物品的两种模型恰好是遵循相同的邻接链而工作的。它们的区别仅是应该使用哪一种“快捷方式”来加速计算。例如,把物品 B 推荐给用户 1 时会参照邻接链用户 1—物品 A—用户 2—物品 B(用户 1 对物品 A 评分,物品 A 又被用户 2 评分,用户 2 又对物品 B 评分)。基于用户的模型遵循了这样的邻接链,并提前计算了基于用户的相似度。通过这种方式,该模型创建了一个“快捷方式”,这个快捷方式绕过了子链用户 1—物品 B—用户 2,而是把这个子链替代成用户 1 和用户 2 之间的相似度。同样,基于物品的方法恰好也遵循了相同的邻接链,但是创建了另外的“快捷方式”,它把物品 A—用户 2—物品 B 替换成物品 A—物品 B 之间的相似度。

我们在这里得到的另一个经验是,把基于邻域的模型当作“基于记忆的”,而把采用矩阵因子分解技术的模型以及爱好当作“基于模型的”并不总是恰当的,至少在使用准确的基于邻域的模型时是不恰当的,因为这些模型差不多与 SVD 一样都是基于模型的。事实上,另一个方向也是如此。更好的矩阵分解模型,如 SVD++ 模型,也是基于内存的,因为在进行在线预测时,它们对内存中存放的所有评分求和,参见规则(5.3)。因此,“基于内存”的技术和“基于模型”的技术这种传统的分类方法在划分本章考察的技术时并不合适。

目前为止,我们集中讨论了基于邻域的模型和矩阵分解模型之间的关系。然而,实际上打破这些关联,并用区别足够大的基于邻域的模型来增强因子分解模型是很有利的,因为这些基于邻域的模型能对因子分解模型进行很好的补充。这样的组合能够提高预测准确度[3, 17]。达到这个目的的关键是使用更加本地化的基于邻域的模型(使用 5.4 节中的模型,而不是 5.5 节中的模型),在这些模型中,近邻的个数是有限制的。对近邻的个数进行限制或许不是构造单独的基于邻域的模型的最好方式,但是该方法却能使基于邻域的模型和因子分解模型有足够大的区别,这样做的目的是增加一种局部性的观点,而这种观点正是全局化的因子分解模型所欠缺的。

## 参考文献

1. Ali, K., and van Stam, W., “TiVo: Making Show Recommendations Using a Distributed Collaborative Filtering Architecture”, *Proc. 10th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, pp. 394–401, 2004.
2. Bell, R., and Koren, Y., “Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights”, *IEEE International Conference on Data Mining (ICDM'07)*, pp. 43–52, 2007.
3. Bell, R., and Koren, Y., “Lessons from the Netflix Prize Challenge”, *SIGKDD Explorations* 9 (2007), 75–79.
4. Bell, R.M., Koren, Y., and Volinsky, C., “Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems”, *Proc. 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.
5. Bennet, J., and Lanning, S., “The Netflix Prize”, *KDD Cup and Workshop*, 2007. [www.netflixprize.com](http://www.netflixprize.com).
6. Canny, J., “Collaborative Filtering with Privacy via Factor Analysis”, *Proc. 25th ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR'02)*, pp. 238–245, 2002.
7. Blei, D., Ng, A., and Jordan, M., “Latent Dirichlet Allocation”, *Journal of Machine Learning Research* 3 (2003), 993–1022.
8. Das, A., Datar, M., Garg, A., and Rajaram, S., “Google News Personalization: Scalable Online Collaborative Filtering”, *WWW'07*, pp. 271–280, 2007.

9. Deerwester, S., Dumais, S., Furnas, G.W., Landauer, T.K. and Harshman, R., "Indexing by Latent Semantic Analysis", *Journal of the Society for Information Science* **41** (1990), 391–407.
10. Funk, S., "Netflix Update: Try This At Home", <http://sifter.org/~simon/journal/20061211.html>, 2006.
11. Gelman, A., Carlin, J.B., Stern, H.S., and Rubin, D.B., *Bayesian Data Analysis*, Chapman and Hall, 1995.
12. Goldberg, D., Nichols, D., Oki, B.M., and Terry, D., "Using Collaborative Filtering to Weave an Information Tapestry", *Communications of the ACM* **35** (1992), 61–70.
13. Herlocker, J.L., Konstan, J.A., and Riedl, J., "Explaining Collaborative Filtering Recommendations", *Proc. ACM Conference on Computer Supported Cooperative Work*, pp. 241–250, 2000.
14. Herlocker, J.L., Konstan, J.A., Borchers, A., and Riedl, J., "An Algorithmic Framework for Performing Collaborative Filtering", *Proc. 22nd ACM SIGIR Conference on Information Retrieval*, pp. 230–237, 1999.
15. Hofmann, T., "Latent Semantic Models for Collaborative Filtering", *ACM Transactions on Information Systems* **22** (2004), 89–115.
16. Kim, D., and Yum, B., "Collaborative Filtering Based on Iterative Principal Component Analysis", *Expert Systems with Applications* **28** (2005), 823–830.
17. Koren, Y., "Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model", *Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.
18. Koren, Y., "Collaborative Filtering with Temporal Dynamics." *Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 447–456, 2009.
19. Koren, Y., "Factor in the Neighbors: Scalable and Accurate Collaborative Filtering", *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(2010):1–24.
20. Linden, G., Smith, B., and York, J., "Amazon.com Recommendations: Item-to-Item Collaborative Filtering", *IEEE Internet Computing* **7** (2003), 76–80.
21. Marlin, B.M., Zemel, R.S., Roweis, S., and Slaney, M., "Collaborative Filtering and the Missing at Random Assumption", *Proc. 23rd Conference on Uncertainty in Artificial Intelligence*, 2007.
22. Oard, D.W., and Kim, J., "Implicit Feedback for Recommender Systems", *Proc. 5th DELOS Workshop on Filtering and Collaborative Filtering*, pp. 31–36, 1998.
23. Paterek, A., "Improving Regularized Singular Value Decomposition for Collaborative Filtering", *Proc. KDD Cup and Workshop*, 2007.
24. Salakhutdinov, R., Mnih, A., and Hinton, G., "Restricted Boltzmann Machines for Collaborative Filtering", *Proc. 24th Annual International Conference on Machine Learning*, pp. 791–798, 2007.
25. Salakhutdinov, R., and Mnih, A., "Probabilistic Matrix Factorization", *Advances in Neural Information Processing Systems 20 (NIPS'07)*, pp. 1257–1264, 2008.
26. Sarwar, B.M., Karypis, G., Konstan, J.A., and Riedl, J., "Application of Dimensionality Reduction in Recommender System – A Case Study", *WEBKDD'2000*.
27. Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., "Item-based Collaborative Filtering Recommendation Algorithms", *Proc. 10th International Conference on the World Wide Web*, pp. 285–295, 2001.
28. Takács G., Pilászy I., Németh B. and Tikk, D., "Major Components of the Gravity Recommendation System", *SIGKDD Explorations* **9** (2007), 80–84.
29. Takács G., Pilászy I., Németh B. and Tikk, D., "Matrix Factorization and Neighbor based Algorithms for the Netflix Prize Problem", *Proc. 2nd ACM conference on Recommender Systems (RecSys'08)*, pp.267–274, 2008.
30. Tintarev, N., and Masthoff, J., "A Survey of Explanations in Recommender Systems", *ICDE'07 Workshop on Recommender Systems and Intelligent User Interfaces*, 2007.
31. Toscher, A., Jahrer, M., and Legenstein, R., "Improved Neighborhood-Based Algorithms for Large-Scale Recommender Systems", *KDD'08 Workshop on Large Scale Recommenders Systems and the Netflix Prize*, 2008.
32. Wang, J., de Vries, A.P., and Reinders, M.J.T., "Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion", *Proc. 29th ACM SIGIR Conference on Information Retrieval*, pp. 501–508, 2006.

## 开发基于约束的推荐系统

Alexander Felfernig、Gerhard Friedrich、Dietmar Jannach 和 Markus Zanker

### 6.1 简介

传统的推荐方法(基于内容的方法[48]和协同过滤方法[40])适用于推荐特性或者口味相似的产品,如书籍、电影或者新闻。但是,在推荐一些如汽车、电脑、房屋或者理财服务的产品的情况下,这些推荐方法就不是最佳的方法了(也可参考第11章)。例如,房屋交易并不是很频繁,所以这让我们很难在一个房屋产品上收集到大量的评分信息(而协同推荐算法正是需要这些评分信息)。同时,获得推荐的用户不会对这些基于已经过时的产品偏好的推荐满意(基于内容的方法正好需要利用这些偏好)。

基于知识的推荐技术利用用户的显式需求,以及产品领域的深度知识来计算推荐,并通过这两种方式来解决上述问题[11]。这些系统高度重视那些不是通过协同过滤和基于内容的方法得到的知识源。与协同过滤和基于内容的方法相比,基于知识的方法没有冷启动问题,因为在每一次推荐的会话中,推荐的需求都是被直接引出的。但是,凡事都有两面性。基于知识的推荐的缺点是所谓的知识获取障碍,知识整理工程师需要花费很大的努力将领域专家提供的知识转化为规范的、可用的表达形式。

具体来说,基于知识的推荐方法可以分为两种:基于样例的推荐[3, 4, 36]和基于约束的推荐[11, 13]。<sup>①</sup>在利用已有知识方面,这两种方法是很相似的:先收集用户需求,在找不到推荐方案的情况下[12, 13, 43],自动修复与需求的不一致性,并给出推荐解释。两种方法的区别在于推荐方案是如何被计算出来的[11]。基于样例的方法通过相似度的基础来决定推荐结果,而基于约束的方法主要利用预先定义好的推荐知识库,即一些描述用户需求以及与此些需求相关的产品信息特征的显式关联规则。本章只关注基于约束的推荐方法的大概内容。至于详细的基于样例的推荐方法,请读者参考[3, 4, 36]。

一个基于约束的推荐系统(见文献[16])的推荐知识库通常定义为两个变量的集合:  $(V_C, V_{PROD})$  和三个不同的约束的集合:  $(C_R, C_F, C_{PROD})$ 。这些变量和约束就是约束满足问题的主要成分[54]。一个约束满足问题的解,包括了具体变量的实例化,例如,所有特定的约束都得到了满足(见6.4节)。

**客户特性:**  $V_C$  描述了客户可能的需求,即需求就是客户特性的实例化。在理财服务领域,客户是否愿意冒风险就是一个客户的特性,冒风险意愿为低表示了客户的一个具体需求。

---

Alexander Felfernig(contact author), Graz University of Technology e-mail: alexander.felfernig@ist.tugraz.at  
 Gerhard Friedrich, University Klagenfurt e-mail: gerhard.friedrich@uni-klu.ac.at  
 Dietmar Jannach, TU Dortmund e-mail: dietmar.jannach@tu-dortmund.de  
 Markus Zanker, University Klagenfurt e-mail: markus.zanker@uni-klu.ac.at  
 翻译: 张思, 沈耀 审核: 严强, 吴亮, 郑州大学-吴宾

① 基于效用的推荐系统也经常被归结到基于知识的推荐方法的类别下,例如文献[4],详细了解基于效用的推荐系统方法请参考[4, 13]。

**产品特性:**  $V_{PROD}$  描述了一个给定产品种类的特性。例如, 推荐的投资周期、产品类型、产品名称、期望收益等都是投资产品的特征。

**约束:**  $C_R$  是对客户需求可能的实例化的系统约束。例如, 短期投资就和高风险投资不相容。

**过滤条件:**  $C_F$  定义潜在客户需求和特定产品种类的关系。例如, 一个缺乏理财经验的客户就不应该接受高风险的投资产品。

**产品:** 最后,  $C_{PROD}$  表示产品特性在可允许范围内的实例。 $C_{PROD}$  代表了析取范式的一个约束, 该范式定义了  $V_{PROD}$  中变量的可能实例上的基本限制条件。

一个简单的理财领域的推荐知识库可用如下的例子表示(见例 6.1):

**例 6.1** 推荐知识库( $V_C$ ,  $V_{PROD}$ ,  $C_R$ ,  $C_F$ ,  $C_{PROD}$ )

$V_C = \{kl_c: [\text{专家, 普通, 新手}] \dots\dots\dots /* 专业等级 */$   
 $wr_c: [\text{低, 中, 高}] \dots\dots\dots /* 风险承担意愿 */$   
 $id_c: [\text{短期, 中期, 长期}] \dots\dots\dots /* 投资时长 */$   
 $aw_c: [\text{是, 否}] \dots\dots\dots /* 是否需要顾问 */$   
 $ds_c: [\text{储蓄, 债券, 股票, 基金, 单股}] \dots\dots\dots /* 直接搜索的产品 */$   
 $sl_c: [\text{储蓄, 债券}] \dots\dots\dots /* 低风险投资类型 */$   
 $av_c: [\text{是, 否}] \dots\dots\dots /* 基金是否可用 */$   
 $sh_c: [\text{股票基金, 单股}] \dots\dots\dots /* 高风险投资类型 */\}$

$V_{PROD} = \{name_p: [\text{文本}] \dots\dots\dots /* 产品名 */$   
 $er_p: [1..40] \dots\dots\dots /* 回报率期望 */$   
 $ri_p: [\text{低、中、高}] \dots\dots\dots /* 风险等级 */$   
 $mniv_p: [1..14] \dots\dots\dots /* 产品最短投资时长(年) */$   
 $inst_p: [\text{文本}] \dots\dots\dots /* 财务机构 */\}$

$C_R = \{CR_1: wr_c = \text{高} \rightarrow id_c \neq \text{短期}, CR_2: kl_c = \text{新手} \rightarrow wr_c \neq \text{高}\}$

$C_F = \{CF_1: id_c = \text{短期} \rightarrow mniv_p < 3, CF_2: id_c = \text{中期} \rightarrow mniv_p \geq 3 \wedge mniv_p < 6,$

$CF_3: id_c = \text{长期} \rightarrow mniv_p \geq 6, CF_4: wr_c = \text{低} \rightarrow ri_p = \text{低},$

$CF_5: wr_c = \text{中} \rightarrow ri_p = \text{低} \vee ri_p = \text{中}, CF_6: wr_c = \text{高} \rightarrow ri_p = \text{低} \vee ri_p = \text{中} \vee ri_p = \text{高},$

$CF_7: kl_c = \text{新手} \rightarrow ri_p \neq \text{高}, CF_8: sl_c = \text{储蓄} \rightarrow name_p = \text{储蓄},$

$CF_9: sl_c = \text{债券} \rightarrow name_p = \text{债券}\}$

$C_{PROD} = \{C_{PROD}_1: name_p = \text{储蓄} \wedge er_p = 3 \wedge ri_p = \text{低} \wedge mniv_p = 1 \wedge inst_p = A;$

$C_{PROD}_2: name_p = \text{债券} \wedge er_p = 5 \wedge ri_p = \text{中} \wedge mniv_p = 5 \wedge inst_p = B;$

$C_{PROD}_3: name_p = \text{普通股} \wedge er_p = 9 \wedge ri_p = \text{高} \wedge mniv_p = 10 \wedge inst_p = B\}$

在这样的推荐知识库上, 给定一个客户需求的集合, 我们就可以计算推荐结果。鉴别出一个满足客户的需求和意愿的产品集合的任务, 我们定义为推荐任务(见定义 6.1)。

**定义 6.1** 一个推荐问题可以定义为一个约束满足问题。 $(V_C, V_{PROD}, C_C \cup C_F \cup C_R \cup C_{PROD})$ , 其中  $V_C$  是一个表示用户潜在需求的变量的集合,  $V_{PROD}$  是描述产品特性的变量的集合,  $C_{PROD}$  是描述产品实体的约束的集合,  $C_R$  是一个表示客户需求的组合的集合,  $C_F$  (又称过滤条件)是描述用户需求和产品特性的关系的约束的集合, 最后  $C_C$  是表示客户具

体需求的一元约束的集合。

**例 6.2** 基于例 6.1 中的推荐知识库，一个推荐任务可以表示为  $C_c = \{wr_c = \text{低}, kl_c = \text{新手}, id_c = \text{短期}, sl_c = \text{储蓄}\}$ 。

基于推荐任务的定义，现在可以介绍一个推荐系统任务的解(一致的推荐)的表示。◀

**定义 6.2** 对于一个推荐任务  $(V_c, V_{PROD}, C_c \cup C_F \cup C_R \cup C_{PROP})$ ，当且仅当  $C_c \cup C_F \cup C_R \cup C_{PROD}$  无任何冲突时，我们定义  $V_c$  和  $V_{PROD}$  中的变量组合为一致性推荐。

**例 6.3** 对于例 6.1 中的知识库和例 6.2 中定义的客户需求，一个一致的推荐是： $kl_c = \text{新手}, wr_c = \text{低}, id_c = \text{短期}, sl_c = \text{储蓄}, name_p = \text{储蓄}, er_p = 3, ri_p = \text{低}, mniv_p = 1, inst_p = A$ 。◀

除了推荐知识库，我们还需要定义接收推荐用户的预期行为的界面接口。为了支持直观的对话，一个推荐界面接口必须是自适应的(见第三部分)。有不同的选择方案来描述用户界面的可能行为。例如，对话可以建模为显式的有限状态机模型[20]，或者可以构建为更加灵活的形式，让用户自己选择他们喜欢的有趣的特性[37]。

本章关注第一种：直接用有限状态机的模型对推荐对话界面建模[20]。状态之间的传递表示为在用户输入上的接受标准。例如，一个专家( $kl_c = \text{专家}$ )对一个理财服务( $aw_c = \text{否}$ )的推荐结果不感兴趣，则自动跳转到  $q_4$ (支持特定技术产品特征的搜索界面)。图 6.1 的有限状态机模型描述了一个理财服务推荐应用的用户的可能行为。

本章的剩余部分组成如下：6.2 节概括开发基于知识的推荐系统的知识获取的概念和推荐过程的定义，6.3 节介绍在一个推荐对话里，引导和积极支持用户的主要技术，6.4 节简要介绍解决推荐任务的方法，6.5 节讨论基于约束的推荐技术的成功应用，6.6 节介绍基于约束的推荐技术的未来研究方向，6.7 节总结本章内容。

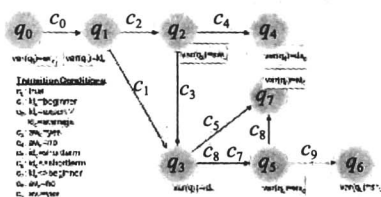


图 6.1 推荐用户界面的描述：一个简单的理财服务的推荐流程。流程从状态  $q_0$  开始，然后，根据用户的知识等级，跳转到状态  $q_2$  或者状态  $q_3$ 。待推荐的物品呈现在最终状态( $q_4$ 、 $q_6$ 、 $q_7$  中的一个)每个状态  $q_i$  都有一个用户特性变量( $q_i$ )来表示在这个状态下，需要提的问题

## 6.2 推荐知识库的开发

在商业应用中，能够对知识工程师和领域专家在开发和维护推荐应用时提供支持，并因而利于尽可能地减小知识获取的瓶颈困难的技术，是成功应用基于约束的推荐技术的主要前提。因为领域专家编程技能有限，通常在开发和维护知识库的问题上，知识工程师和领域专家之间存在着差异[13]。这样一来，领域专家只需负责提供知识规则，而不需要将其形式化，并将其转为成一种可执行的表达方式(即推荐知识库)。

文献[13]中介绍的 CWAdvisor 环境的主要目的就是要减少上面提到的知识获取的瓶颈：它为领域专家提供支持，实现了知识库的开发和维护的流程自动化。为了展示其如何工作，我们将在接下来的部分介绍 CWAdvisor 环境的部件。CWAdvisor 知识获取环境(CWAdvisor Designer)所遵循的设计原则，对知识的获取和维护，起到了至关重要的作用[8, 13]。

- 第一，快速原型流程支持具体性原则，在这样的原则下，用户可以快速检查对文本解释、产品特性、图片、推荐过程定义和推荐规则的修改带来的效果。这样的功能通过模块化的形式来实现，在这样的形式下，我们可以将图形定义的模型特性直接转换为可执行的推荐应用。



- 第二, 所有的信息单元的改变都可以通过图形表达出来。这样的功能允许领域专家在缺乏技术背景的情况下, 能更轻松地接受和使用知识获取环境。领域专家从编程细节的技术困境中解放出来, 这样的方法遵循了将应用逻辑和实现细节严格分离的原则。
- 第三, 集成的测试和调试环境遵循了立即响应的原则: 推荐知识库和推荐过程中的错误定义都能被自动地检测和报告(支持终端用户调试)。这样, 知识库就被结构化地组织起来, 直到所有满足知识库的测试样例都得到满足, 它才会在生产环境下发生作用。这导致的一个直接结果是, 错误的结果被特殊处理, 因而推荐结果的可靠度得到提升。

图 6.2 提供了一些 CWAdvisor 推荐开发环境的主要建模概念的实例[13]。这个环境可以用来设计推荐的知识库(参考例 6.2), 即客户特性( $V_C$ )、产品特性( $V_{PROD}$ )、约束( $C_R$ )、过滤条件( $C_F$ )、产品种类( $C_{Prod}$ )可以具化到图形层面上。图 6.2 的上面部分展示了过滤条件( $C_F$ )的设计界面, 下面部分展示了面向上下文的通用性约束的界面。图 6.3 展示了 CWAdvisor Process Designer 的用户界面。这些组件使得推荐流程的图形化设计变为可能。给定了这样的流程定义, 推荐结果就能自动化地生成(见图 6.4)。

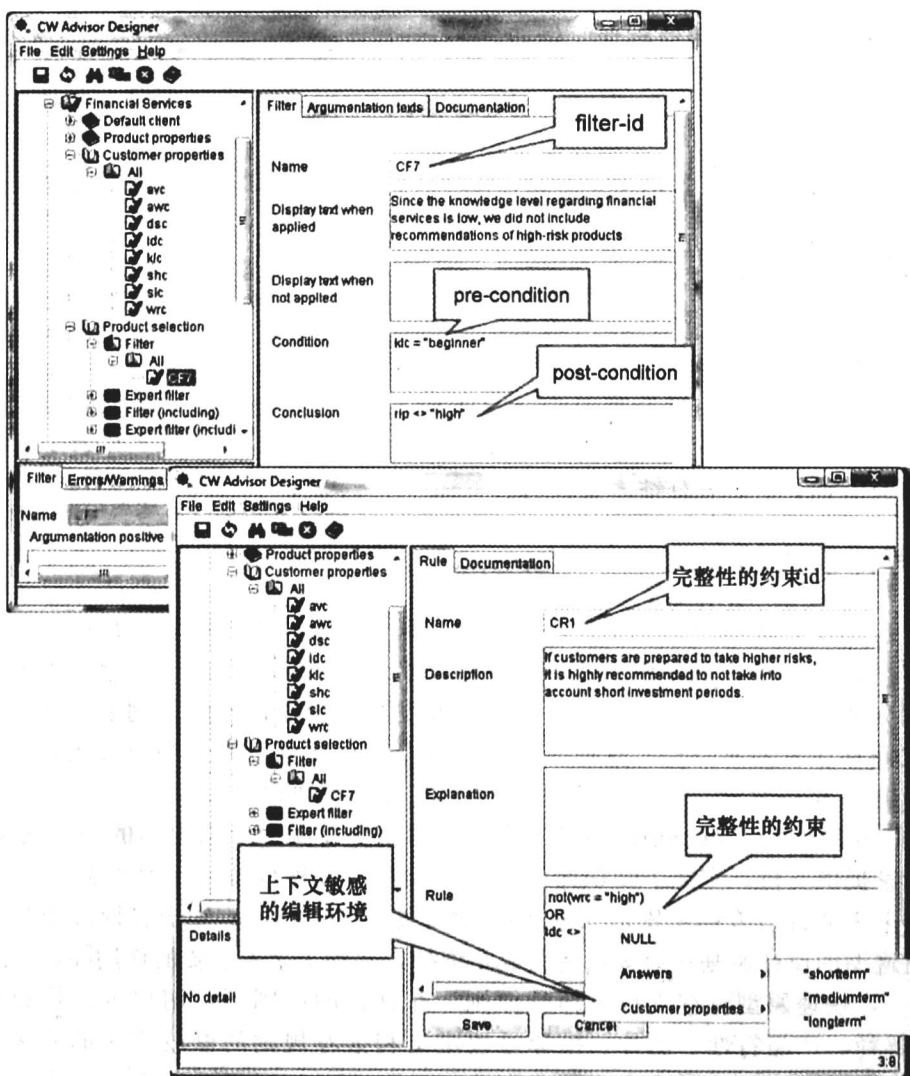


图 6.2 CWAdvisor 设计环境。过滤条件和完整性约束都能在上下文敏感的编辑环境中定义

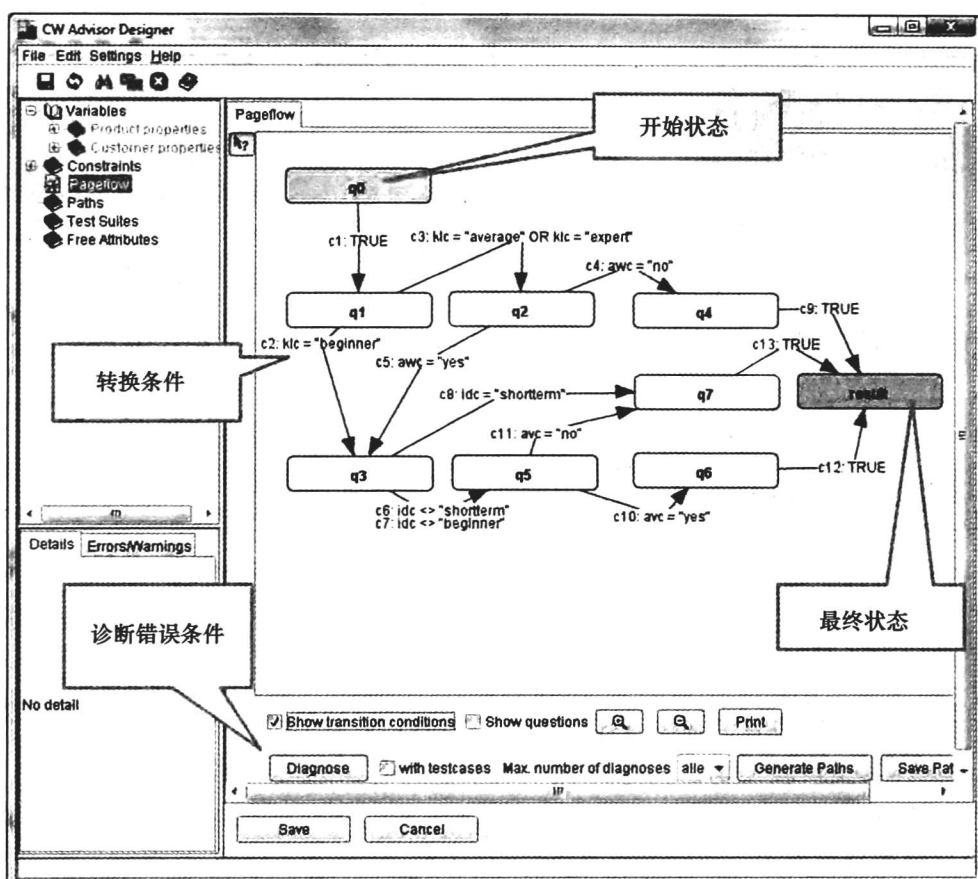


图 6.3 CWAdvisor 设计环境。推荐流程可以在图形化的层面来详细说明并且能够自动地转换到相应的可执行的表达式形式。在基础的基于模型的识别器上[20]，错误的转换条件能自动地被识别

有时候，推荐流程是错误的，例如，状态间转换方式定义不恰当的情况下，将导致一个推荐会话不能成功完成。在图 6.1 中，如果我们改变转换条件，将  $c_1: klc = \text{beginner}$  改为  $c'_1: klc = \text{expert}$ ，缺乏理财服务领域相关知识的用户将不会被引导到后续状态( $q_2$  或  $q_3$ )。在更复杂的流程定义下，人工识别和修复这样的错误是单调乏味的工作，并且容易出错。文献[20]介绍了一种方法，能够帮助我们自动地识别和修复这样的错误状态。这个方法的概念基础是基于模型的诊断方法[20]，这样的方法有利于找出引发错误转移的条件因素的最小集合。

除了图形化的流程定义，CWAdvisor 设计器支持测试样例的自动化生成(输入包含推荐产品的序列)[16]。一方面，这样的测试样例能用于回归测试，例如，可以在推荐应用投入生产环境前使用这些测试用例进行测试。另一方面，测试样例能用于推荐知识库的错误调试(如果有一些测试样例不能得到满足)和错误流程定义(如当推荐流程卡住的时候)。

例 6.4<sup>⊖</sup>介绍了推荐知识库调试的基本原则[10, 12, 13, 16]，对推荐过程定义自动化调试有兴趣的读者可以参考文献[20]。识别推荐知识库错误的一个典型方法就是用一个包含了一系列测试样例的测试集( $e_i \in E$ )进行测试。为了简单起见，我们假设领域专家只

⊖ 简单起见，这里省略了  $V_{PROD}$ 、 $C_F$  和  $C_{PROD}$  的说明。

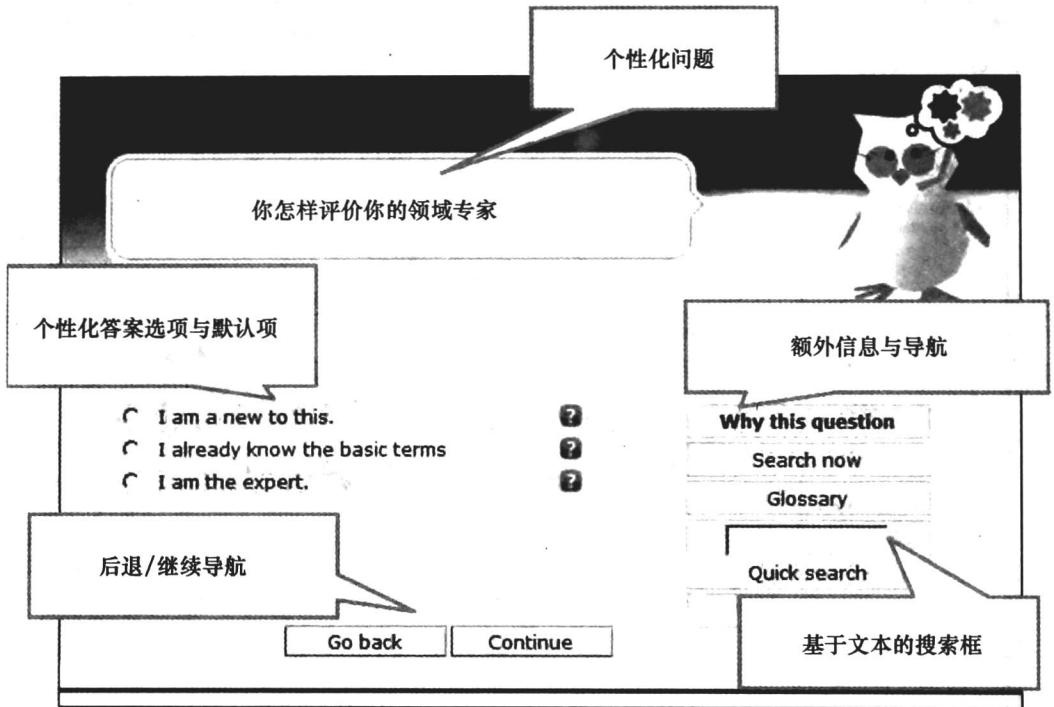


图 6.4 交互式的个性化偏好提取示例。用户通过回答问题来指定他们的偏好

提供了一条样例,  $e_1: wr_c = \text{高} \wedge rr_c \geq 9\%$ 。测试  $e_1 \cup C_R$  的结果是空集, 因为  $e_1$  和  $C_R$  不一致。更详细地观察例子, 我们可以发现  $CR_2$ 、 $CR_3$  和  $e_1$  都不一致。 $CR_2$ 、 $CR_3$  被定义为冲突集[33, 45], 我们只需要删除其中一个元素就可以解决不一致问题。冲突集的计算可以通过文献[33]提供的冲突检测算法来实现, 文献[20]详细介绍了冲突问题的自动解决。

**例 6.4** 错误的知识推荐库( $V_C, V_{PROD}, C_R, C_F, C_{PROD}$ )

$V_C = \{rr_c: [1 \sim 3\%, 4 \sim 6\%, 7 \sim 9\%, 9\%] \dots \dots \dots /* \text{回报率} */$   
 $wr_c: [\text{低}, \text{中}, \text{高}] \dots \dots \dots /* \text{是否愿意承担风险} */$   
 $id_c: [\text{短期}, \text{中期}, \text{长期}] \dots \dots \dots /* \text{投资时长} */\}$

$C_R = \{CR_1: wr_c = \text{中} \rightarrow id_c \neq \text{短期}$   
 $CR_2: wr_c = \text{高} \rightarrow id_c = \text{长期}$   
 $CR_3: id_c = \text{长期} \rightarrow rr_c = 4\% \sim 6\% \vee rr_c = 7\% \sim 9\%$   
 $CR_4: rr_c \geq 9\% \rightarrow wr_c = \text{高}$   
 $CR_5: rr_c = 7\% \sim 9\% \rightarrow wr_c \neq \text{低}\}$

$V_{PROD} = \{\} C_F = \{\} C_{PROD} = \{\}$

在从构建理财服务[18]、电力设施[13]、电子旅游[74]等领域的商业项目中得到的经验来看, 以上提到的原则对于知识获取和环境维护的设计起到了至关重要的作用。文献[10]对用户研究发现, 因为开发、测试、调试等工作都可以在图形化的界面里完成, 这大大节约了用户的时间。从理财服务领域[18]的经验来看, 最初的知识库必须要领域专家和技术专家(即知识工程师)的合作配合来完成。在那之后, 大多数开发和维护工作都可以直接由领域专家来完成(例如, 更新产品列表、修改约束、定义推荐过程等)。

### 6.3 推荐过程中的用户导向

基于约束的推荐系统的运行建立在用户的需求和愿望能明确表述的前提下,因此这些与用户需求相关的知识必须在推荐系统建立之前就得提前做好。对于这样的需求获取过程,它的实施复杂度是递增的。通用的办法包括以下几点:

- 与会话无关的顾客信息:用户确认他们的偏好以及兴趣设置,例如,通过指定他们的一般兴趣范围(见第22章)。这是一个在门户网站和社交平台通用的办法。
- 每个会话的静态填写表格:每当顾客使用一次推荐系统时,他们便会填写一个静态的网络表单。这样的界面很容易实现,而且互联网用户也非常熟悉这样的界面,因此它们经常被用在网络购物的搜索之中。
- 会话式的推荐对话框:在一个交互式的对话中,举例来说,基于“评判”的[8](见第13章)、“向导式的”和基于表单的偏好提取的对话[13]、自然语言交互[59]或者这些技术的整合,推荐系统不断地收集着用户的偏好信息。

在基于约束的推荐系统的上下文中,尤其是最后一类的偏好提取扮演着非常重要的角色,而且也将是本章讨论的重点。因为在类似理财服务[18]、电子生活消费品[25]等综合领域中,推荐建议往往会给予推荐系统相互作用的终端用户带来显著的认知负担。因此,为了确保这套系统能为广大的在线用户社区所使用,充足的用户交互是必不可少的。

当然,在用户自己填写的个人资料中,这些静态信息也可以当作基于约束的推荐系统的输入源。将这些笼统的信息(包括详细的人口统计信息)整合入推荐系统的步骤也是简单明确的。然而在很多情况下,这样的信息很容易是不明确的或空泛的,因此,对一个基于详细知识的推荐过程而言,这些信息碎片的实用性是很有限的。

基于上述提到的原因,静态填充表格在某些应用上的表现很好。但是,在知识密集型领域,基于约束的推荐系统会经常生成,这样的方法可能会过于简单,尤其是因为在线用户群可能因他们的技术背景不同而不同,因此,对所有用户都问同一套问题或者都在同一技术细节的水准上,是不合适的[25]。

最后,在本章中依然不会聚焦于自然语言交互。因为只有非常少的例子,如文献[59]中,运用了(作为补充的)自然语言推荐系统用户界面。尽管随着自然语言处理领域的发展和虚拟拟人顾问在不同的网站以扩展组件的形式出现,但由于一些不同的原因,它们至今几乎没有被用来向用户推荐物品。首先,这类的对话往往是用户主导的基于约束的推荐系统会经常生成,例如,用户将积极主动地提出问题。但在综合领域,尤其是新用户,无法系统地提出这样的问题,如正确的中期投资策略。其次,由于这类系统能够进行随意的会话,它们的知识习得的代价是相当高的。最后,终端用户在交互后常常把更多的智能归功于这样的拟人替身,而不是归功于伴随有使他们失望的风险的系统。

**评论** 对基于知识的推荐系统,评论是一个普遍的交互方式。它第一次被提出是在文献[6]中的类似会话推荐的基于案例推理(CBR)的上下文场景中。这样的理念是为了展现每件商品(实例),例如,数码相机或理财产品,依照各个产品的评价,对产生交互的用户提供反馈。例如,一个用户可能需要一个“短期投资”或“低风险”的理财产品。这个推荐一核实一修正的闭环不断地重复,直到期望得到的物品被找到。需要注意的是,尽管这样的模式是为CBR推荐方式<sup>⊖</sup>而开发的,它仍然能被运用于基于约束的推荐系统,因为评价能

<sup>⊖</sup> 用评论后续实例的方法探索数据,这种想法在20世纪80年代的信息检索领域[55]已经被提出过。

够直接被转化成附加约束,从而在某些特征上反映出用户的方向性偏好。

与在很多在线商城都能找到的详细搜索表单相比,评论的交互方式有它的优势,因为它能够支持用户交互式地搜索商品库。此外,这种通常被称为微调的方法,对于新用户而言也很容易理解。可是,开发一个评论程序,需要一些领域知识,如用户反馈的特征集,适当地量化属性增量,或者枚举域内符合逻辑的属性顺序。此外,当需要将用户需求映射到产品特征上去时,额外的工作量也是不可避免的。

基本的评价图表后来也扩展为支持**复合评论**[44, 52],即在一个交互周期内,用户能对多个特征给出反馈。在理财服务领域,用户也因此只需一步就能找到一个更低分险、更长投资期限的产品,这样也就减少了所需的交互周期次数。在有些提前设计的复合评论[6]中已经是合理的初始建议的同时,在[44]中也提出另一种意见:合理的评论应该是被动态决定的,且应取决于当前用户的类目空间中剩余的物品,尤其是这些剩余物品的变化差距。实验评测结果表明,这样的复合评价能帮助我们显著地降低所需的交互周期次数,从而使整个交互过程更加有效。此外,实验还表明,复合评论(即使被限制大小,用户仍然能理解)也能帮助用户理解系统产生推荐的逻辑。

关于评论研究的最新进展,包含了精致的可视化界面的作用[62]、手机推荐系统中的应用推荐方法[74]、关于决策精度和认知效果评论样式的评估[20]。

**个性化偏好提取对话** 在基于约束的推荐系统中,另一种获取用户愿望和需求的形式依赖于明确模式化并自适应地偏好提取对话。这样的对话模型,可以直接用**对话框形式**[2]表示,或者是跟 CWAdvisor 系统中一样用有限状态自动机表示[20, 13]。

在之后的系统中,在推荐结果展示之前,终端用户由“虚拟顾问”通过一系列关于特定需求的问题来引导,见图 6.4 中的会话样例。与静态填充表格对比,这套问题集是个性化的。也就是,根据当前的场景和以前的用户回答,系统会提出一套不同的问题集(或许也用了一种不同的专业术语或非专业术语[29])。

在 CWAdvisor 系统中,用户界面的变化是通过有限状态自动机建立在手动设计的个性化规则和明确的对话模型之上的,如图 6.1 所示。因此我们选择了一种方法,作为网站用户熟知的填充表单和基于自然语言处理方法的充分自由的会话之间的折中。

从技术上讲,通过那些常常被用来捕获用户需求的约束变量,在图 6.1 中的有限状态机的顶点以逻辑表达式表示。通过面向终端用户的图形流程建模编辑器,处理会话和个性化模型的过程由 CWAdvisor 系统支持。在运行时,这个架构的交互处理组件收集了用户的输入并评估转移条件,以便决定如何继续对话,更多信息请参考[13]。

除了个性化对话,其他在内容、交互、呈现等方面的不同自适应形式也运用在系统中[30],目的是尽可能好地支持终端用户的偏好抽取和对话解释的功能设计。

虽然高度动态的和自适应的网络应用程序在易用性和用户体验上有价值,但在技术实现上是有挑战性的,尤其是在基于约束的推荐下维护有这种弹性的用户界面。在这个背景下的主要问题是在“模型”“浏览界面”和自控逻辑之间的强连接,例如,想象这样的情形:对话模型需要掺入一个新的问题(变量),一个新的回答建议(新的变量值域),或整个对话页面(新的对话状态机)。在所有情况下,用来呈现来自推荐应用“界面”的网页,都必须做出相应的改变。因此,个性化偏好提取工作的工具包,必须提供至少部分自动化更新用户界面的机制,在 CWAdvisor 中的基于模板的方法的详细信息请参考文献[30]。

**处理无法实现的或太宽泛的用户需求** 在基于约束的推荐系统中,在个性化偏好提取的场景下开发这样的用户接口不是唯一的挑战性难题。在下文中,我们将简述这项技术在



实际应用遇到的其他难题(见第 15 章和第 16 章)。

在基于约束的推荐系统中,常常遇到目录中没有找到满足用户所有约束条件的物品的场景。在一次交互式推荐系统的通信中,像“没找到匹配的商品”这样的信息无论如何都是不受欢迎的。因此而产生的问题是,如何处理这样的情况,即在许多情况下至少最初依赖一些查询机制来从产品目录(范例库)中检索出一套初始案例集,它同样发生在基于 CBR 的推荐系统中。一个在基于 CBR 推荐系统的场景下提出的可行处理方法是基于宽松查询[39, 43, 47, 23, 27]。在 CBR 推荐系统的场景下,可推荐物品集合是被概念化地存储在一个数据库表里;实例检索包含了发送一个(源于用户需求的)联合查询  $Q$  到这个实例检索中的过程。然后宽松查询指向寻找一个原始查询  $Q$  的(最大)子查询  $Q'$ , 最终返回至少一个物品。

一般的查询放宽技术也能应用到基于约束的推荐系统之中。考虑例 6.5(改编自[27]), 4 个物品  $C_{PROD}$  的目录在图 6.5 中以表格的形式展示。

$name_p$	$sl_p$ (低风险投资类型)	$ri_p$ (相应等级)	$minv_p$ (最短投资周期)	$er_p$ (期望收益)	$inst_p$ (理财机构)
$p_1$	股份基金	中	4	5%	ABank
$p_2$	单股	高	3	5%	ABank
$p_3$	股份基金	中	2	4%	BInvest
$p_4$	单股	高	4	5%	CMutual

图 6.5 样例物品目录(金融服务)

#### 例 6.5 查询放宽

为了样例的清晰度和简洁度,在一个咨询应用的“专家栏目”上,假设顾客能直接指定想要投资的产品特征。因此,相应的客户资产集  $V_C$  包含  $sl_c$  (投资类型)、 $ri_c$  (风险等级)、 $minimum\_return_c$  (保底收益值)和  $investment\_duration_c$  (预期投资时间)。在这个样例中,过滤约束(条件)简单地把顾客需求从  $C_c$  映射到物品特征,也就是  $C_F = \{CF_1 : sl_c = sl_p, CF_2 : ri_c = ri_p, CF_3 : investment\_duration_c \geq minv_p, CF_4 : er_p \geq minimum\_return_c\}$ 。让具体的顾客需求  $C_c$  变换如下:  $\{sl_c = singleshares, ri_c = medium, investment\_duration_c = 3, minimum\_return_c = 5\}$ 。

显而易见,在给定的任务中,目录里没有任何物品(见图 6.5)满足所有的相关约束条件。当遵循“约束放宽”原则时,当前的目标包含了寻找一个推荐系统能找到的符合约束条件  $C_F$  的最大子集。最大化的标准则是选取有代表性的,因为约束条件直接涉及顾客需求,也就是说,更多的约束条件被保留时,就会有更好的补偿物品来匹配这些需求。

当第一次发现寻找  $C_F$  确立一致性子集的问题看起来并不是那么复杂时,在实际操作中,计算效率却变成了瓶颈。给定一个包含  $n$  个约束条件的基本约束,所有可能的子集数是  $2^n$ 。由于现实可用的推荐系统不得不并行服务许多用户,并且一般情况下可接受的响应时间大约是一秒,所以朴素的子集搜索是不可取的。

为了解决这个问题,人们提出了各种不同的技术。在文献[39]中,举例来说,在 CBR 推荐系统中,提出的方法是一个从失败查询中恢复的增量混合法。在文献[47]中,提出了基于手动定义特征分类放宽的方法,尽管它的不完备性在旅游业推荐系统中已被证明是一个有效的解决方案。最后,在文献[27]和文献[26]中,在基于约束的推荐系统中,已经开发了查询放宽问题的完整算法集。这些算法不仅支持线性时间(以预处理过程和少量增量内存需求为代价)的最小放宽计算,还支持导致“至少  $N$ ”件剩余商品的放宽计算。此外,交互式的、

增量查询放宽的冲突导向算法也被提了出来，它使用了最新的冲突检测技术[33]。

线性时间约束放宽技术的主要思想简单描述如下。代替了尝试约束组合，它单独评估相关约束，并给每一个约束条件分配数据结构，同时商品目录的清单满足这些约束条件，见图 6.6。

ID	Product $p_1$	Product $p_2$	Product $p_3$	Product $p_4$
$CF_1$	0	1	0	1
$CF_2$	1	0	1	0
$CF_3$	0	1	1	0
$CF_4$	1	1	0	1

图 6.6 单独评估子查询。比如，在假设  $sl_c = \text{singleshares}$  的情况下，过滤条件  $CF_1$  将产品  $p_1$  过滤

表格可以理解如下。投资型(单股)约束条件  $CF_1$  在表格第一行将过滤掉物品  $p_1$  和  $p_3$ 。

给出这个表格后我们能轻松地断定，给定  $C_F$  集的约束条件必定是不严格地会有某一物品出现在结果集中，也就是说，与约束条件和用户需求一致。例如，为了让  $p_1$  出现在结果集中，约束条件  $CF_1$  和  $CF_2$  必须要被放宽。让我们称为  $p_1$  的“特定商品放宽”。来自[27]中的方法的主要思想是，对于给定的商品  $C_{PROD}$  的所有“最佳”放宽，过滤条件  $C_F$  和给定的具体需求集  $C_c$  必须在特定商品放宽列表之中。因此，扫描特定商品放宽的集合就足够了，也就是说，在这个阶段中并不需要更深层次的约束解决步骤。

在这个样例中，因为只有用户的一个需求被忽略，所以当放宽的约束条件数量决定最佳选择时，约束  $CF_2$  的放宽是最优的。其他所有的放宽要求至少忽略两个约束条件，这样才能通过统计每一列中零的个数来简单地决定。请注意，相关约束条件的数量只是可行的最优化准则之一。考虑每个用户的附加“妥协成本”的最优化准则，也能在这个技术基础上应用，只要满足损失函数值是根据放宽程度而单调递增的。

从技术上讲，特定商品放宽的计算能够通过位运算非常高效地完成[27]。此外，在推荐系统的初始化时，也能先预处理部分计算过程。

**对于无法实现的需求提供备选方案** 在一些应用领域，自动化的或交互式的个别约束放宽或许不能帮用户摆脱他的需求无法满足时的情景。考虑这样的情景，比如，在交互式放宽方案中，推荐算法提出一系列约束备选方案集。假设用户接受其中的一个建议，也就是说，同意放宽与  $V_c$  的两个变量  $A$  和  $B$  有关的约束条件。然而，如果  $A$  和  $B$  的值对用户而言是重要的(或强制性的)，则之后将会把不同的约束条件加在这些变量之上。但是，这些新值会再次导致与该用户其他需求的不一致性。最终这可能导致一个不合理的情况，即用户终止尝试不同的取值但也得不到明确的建议，它衡量了接收一个一致性推荐建议的价值。

总的来说，它会因此而变得让用户满意，设想系统能马上为  $A$  和  $B$  的新值提出建议，当这个用户的其他需求也被更进一步地考虑时，保留在结果集里的一些物品得到了保证。

让我们先考虑运用在[39, 43, 47]中的基础 CBR 式样案例检索问题，在这里约束条件被直接强加于物品特征之上。这个案例中的约束条件应该是  $\{sl_p = \text{singleshares}, ri_p = \text{medium}, minv_p < 3, er_p \geq 5\}$ 。此外，依然没有物品满足这些需求。

在这样的情况下，有关物品类目的详细信息常常被用来计算一组针对可供选择的约束条件(“恢复”)个体特征的建议。基于这个信息，系统能够——代替只建议用户放宽投资类型和

投资持续时间的需求——告知用户“即使单股需求被抛弃，并且最小投资持续时间设定为4”，也能找到一个或多个物品。因此，用户能(非有效地)预防尝试最低投资持续时间。

在这个案例中，此类替代值的计算可由系统通过选择一个放宽的替代物和搜索满足剩余约束条件的商品类目来完成。投资持续时间和投资类型的取值(比如，图6.5中物品1的值)也能直接被选取为终端用户的建议[19, 14]。

当这种方法看起来既直观又简单时，在实际应用中却需要处理如下问题。

- 可用备件的数量。在实际情况下，可用的备件替代量往往是非常大的，因为每一种可能的放宽——这里可能就已经有很多——都存在着各式各样的解决方案。在实践中，不管怎样，终端用户无法接受过多的备选方案。因此，问题转换成了如何选择并排序这些修正过的备选方案。
- 修正建议的大小/长度。修正对包含超过三个特征替代值的建议，也不容易为终端用户所理解。
- 非平凡约束的计算复杂度。当只允许简单的商品特征约束条件时，类目信息有利于制订上述可行的修正方案。在像 CWAdvisor 这样的基于约束的推荐系统中，不管怎样，经常涉及定性用户需要的(技术)产品特征的约束定义是可行的。因此，修正的建议也必须与用户需求相关，这意味着可行修正备选方案的搜索域是由用户相关的变量域所决定的。此外，决定是否为用户需求(也就是，一个修正替代品)的特定组合导致的非空解集，可能需要高代价的类目查询来实现。

为了至少在一定程度上定位这些问题，在理财服务应用中，针对修正建议的计算，CWAdvisor 系统使用了查询放宽组合、不同的启发式搜索以及额外的特定领域知识相结合的方法[17]。

运用在系统中的这种方法通过修正备选方案的有界查询、交叉放宽查询。合适的放宽是由它们的技术递增顺序决定的。对于每一个放宽，修正备选方案是通过改变那些包括放宽的约束变量值来决定的。备选方案取值的选择，可以以“近似”启发式搜索为导向，它是以外部的或内在的确切顺序为基础的。因此，例如，一个用户的需求为“至少希望收益5%”，则邻域值“4%”会被评估，假设这样的一个备选方案对终端用户而言比一个更强放宽更容易接受。为了避免太多的相似修正建议，算法以几个临界值为参数，比如，决定一个放宽的修正次数，一次放宽的最大上限等。总的来说，在理财服务领域的证据表明，这样的修正特征，即使它是基于启发式的、作为缩短所需对话长度的一种方法，也得到了终端用户的高度认可。

**查询紧缩** 在一个交互式推荐系统中，除了结果集中没有任何物品，结果集中有太多的物品也不是令人满意的。在一些实际的应用中，用户被告知“找到了太多物品”并需要指定更精确的搜索约束条件。结果往往是只有前面的少数结果得到展示(比如，为了避免过长的页面加载时间)。不管怎样，这样的选择对当前用户而言不是最理想的，因为这样的结果往往是基于类目词条字典序的简单排序。

为了在这种情况下也能更好地服务用户，在文献[48]中，一个基于 CBR 推荐系统的交互查询管理方法被提了出来，它也包含了“查询紧缩”的技术。被提出的紧缩算法把一次输入当作一次查询  $Q$  和它的大型结果集，并挑选出(根据信息理论的约束条件和熵方法)三个特征展现给用户，并建议用户尽可能地精炼查询条件。

总的来说，对于在旅游推荐系统中同时使用了查询放宽和查询紧缩[50]的交互查询管理的评测结果，放宽特征得到了终端用户的高度认可。评测还表明，通过考虑紧缩功能，

对于有能力自行精炼查询条件的终端用户，查询紧缩并不是那么重要。因此，在[41]中提出了一个不同的特征选取方法，它同样把特征流行度的概率模型纳入考虑之中。评估表明，在特定环境下[41]的方法表现更优秀，因为它更容易被终端用户接受并作为进一步精炼查询条件的方法。

## 6.4 计算推荐结果

根据我们对于一个推荐任务的描述(见定义1)，我们现在将讨论相应问题的解决方案。一个典型的解决推荐任务的途径是约束满足算法[54]和结合数据库查询[46]。

**约束满足** 约束满足问题的解决方案寄希望于使用结合回溯法和约束传播法的搜索算法基础——所有概念的基本原理会在下文中解释。

**回溯法** 在每一步，回溯法选择一个变量并把所有可能的值分配给这个变量。它用已分配的和已定义的约束集合来检查分配的相容性。如果当前变量的所有可能的值与当前的分配和约束条件都不一致，约束求解器由原路返回，这意味着之前遍历的变量又被选中了一次。当这次分配的相容性得到保证时，回溯算法的递归过程开始执行，同时下一个变量被选中[54]。

**约束传播法** 纯粹的基于回溯法搜索的主要缺点是，即使不存在解，部分“无效”的搜索域也会被多次搜索。为了使约束解决过程更加高效，约束传播技术被引入。这些技术尝试着修改一些当前的约束满足问题，以便使搜索域能显著缩小。这种方法尝试着创建一个局部相容的状态，来保证多组变量值间也能始终相容。上述改进步骤将一个当前的约束满足问题转变成等价的问题。一种众所周知的本地相容类型是弧相容[54]，即对于两个变量  $X$  和  $Y$ ，当  $Y$  在  $X$  中没有对应的相容的值时， $Y$  的定义域为空。因此，弧相容直接被定义为，如果  $X$  与  $Y$  是一致的，反转后未必是一致的。

当使用约束求解器时，约束被典型地描述为对应编程语言的表达形式。目前许多的约束求解器是在 java 的基础上实现的。(样例可参考 [jacop.osolpro.com](http://jacop.osolpro.com)。)

**联合数据库查询** 联合查询的解决方案是在数据库查询的基础上计算的，它设法检索出满足顾客所有需求的物品。有关数据库技术和数据库表单查询执行的详细信息，请参考文献[46]。

**排序物品** 给定一个推荐任务，约束求解器和数据库引擎都设法确定满足顾客给定需求的物品集。一般情况下，我们需要处理不止一个物品是推荐结果的一部分的情况。在这样的情况下，物品(商品)需要在结果集中排序。在两个案例(约束求解器和数据库引擎)中，我们应用多属性效用理论(MAUT[56])的概念来辅助确定结果集中每个物品的排名。MAUT 的应用样例可参考文献[13]。

可选择的与联合查询结合的 MAUT 应用是概率数据库[35]，它允许在查询中直接指定排序标准。例 6.6 展示了这样的查询，它通过 WHERE 子句挑选出那些满足标准的商品，并按照相似性(在 ORDER BY 子句中定义)排序。最终，取代了用 MAUT 结合上述标准的约束求解器，我们以软约束的形式表述一个推荐任务，变量的每一种组合的重要性(优先级)是在相应通用操作(细节请见文献[1])的基础上决定的。

### 例 6.6 概率数据库的查询

Result=SELECT \*

/\* 计算一个结果 \*/

FROM Products

/\* 从“Products”中选出物品 \*/

WHERE  $x_1=a_1$  and  $x_2=a_2$

/\* “必须”条件 \*/

ORDER BY score(abs( $x_3 - a_3$ ), ..., abs( $x_m - a_m$ )) /\* 基于相似性的通用函数 \*/  
 STOP AFTER N; /\* 答案(结果集)中最多 N 个商品 \*/ ◀

## 6.5 项目和案例研究的经验

CWAdvisor 系统在 2002 年实现了商业化, 自此被超过 35 个不同的应用中所使用。它们已经覆盖了从理财服务[17]到电子消费品或旅游应用[32]等商业领域。在认为推荐系统不擅长的应用领域, 如提供商业计划咨询服务[28], 或者支持软件工程师筛选合适软件评价方法[43], 也得到了很好的发展。

在这样的基础上, 不同形式的经验主义研究已经兴起, 他们尝试着评估基于推荐系统的知识的影响力和商业价值, 也找到了改进他们最先进技术的机会。在下文中, 在他们的研究设计深入用户研究、历史数据评估和生产系统的案例研究的基础上, 我们将对他们进行区分。

**实验用户研究** 模仿真实用户交互, 并探索不同假设下的结果是接受还是拒绝。文献[15]引出了一项研究来评估会话式的、基于知识的推荐算法特定函数的影响, 如推荐解释、推荐的修正方案或商品比较。这项实验随机地把用户分配到不同版本的推荐系统中, 它改变功能函数并应用前置或后置交互调查, 来确定用户的专业知识等级、他们的信任度或推荐系统的胜任感。非常有趣的是, 这项实验表明参与者肯定了这些特定功能函数, 因为这增加他们的专业知识感知水平和对推荐结果的信任度。

COHAVE 项目发起一系列的调查, 来研究心理学理论如何能解释用户在线选择情况下的行为。例如, 非对称显性效应上升, 即使所推荐的物品集包含诱饵商品, 这些商品由于它们的相似性而被其他商品控制, 但是有较低的全局效用。在如电子消费品、旅游和理财服务等领域, 一些用户研究显示, 一个考虑了这些影响的推荐系统, 能和购买决策中的用户信任度一样, 增加一些特定物品的转换率。

**基于历史数据的算法评价** 一份包含以往用户交易的数据集被分成训练集和测试集[25]。训练集用来学习一个模型或调整算法的参数, 目的是使推荐系统能在测试集上预测用户的历史行为。这样的评价方案使算法性能的对比研究成为可能。当协同和基于内容的推荐范例在文献中被广泛地评价时, 基于知识的推荐算法和其他的推荐范例的对比在以前却只引起很少的注意。原因之一是他们难以比较, 因为他们需要不同类型的算法输入: 协同过滤典型地利用用户评分而基于约束的推荐系统却需要明确的用户需求、类目数据和领域知识。所以, 包含所有类型的输入数据集——像由 Burke[14]提供的主要数据集——允许这样的比较, 尽管他们是非常稀疏的。少数中的一个数据集在文献[61]中有描述。这个数据集来自一个提供优质雪茄的零售商, 它包含了代表用户购买行为、输入会话式推荐的用户需求和带有商品详细描述的商品类目的隐式评分。然后, 通过离线试验将利用用户需求的基于知识的算法变种与利用评分的基于内容的协同过滤作比较。其中一个很有趣的结果是, 在以类目覆盖率衡量的惊喜度上, 基于知识的推荐系统的表现并不比协同过滤差。如果基于约束的推荐系统与像 CWAdvisor 系统这样的基于效用的物品评分机制相关联, 这是格外正确的。但是, 如果有 10 个或给更多的来自用户的评分, 协同过滤在准确度上做得更好。虽然如此, 基于知识的推荐系统的评估常常只能衡量编码知识库的质量和它本身的推论。

另一项研究在[60]中功能化了, 它聚焦于明确的用户需求作为唯一的个性化输入机制的情况。它对比了基于知识和协同过滤的不同杂交变种, 协同过滤认为明确的需求是评分



的另一种形式。如果用户明确地表达了一些特殊需求,基于知识的推荐系统的结果集被证明是非常精确的。但是,当只有少量约束请求而且结果集很大时,排序函数不总能确定最佳匹配的物品。与此相反的是,协同过滤学习了需求和实际购买物品之间的关系。因此,这项研究表明,基于知识的推荐系统移除了基于硬性条件的替补者,并且协同过滤算法来评分时,串联策略表现得更好。

因此,在文献[57]中,基于知识和协同过滤之间的分层混合方法被提了出来并得到了验证。在这里,协同过滤学习把用户需求映射到已购物品类目属性的约束条件,并把它们作为输入传给以基于知识为主的推荐系统。在历史数据上的离线试验提供了初步证据,在算法的准确率上,这样的方法能胜过来自领域专家的知识库。在这些基本满意的结果的基础之上,从历史交易数据中自动提取约束条件的研究将会进一步展开。

**生产系统中的案例研究** 是最实际的评价形式,因为用户在真实环境下产生行为并控制内在动机来使用这个系统。在文献[13]中报道了两个来自理财服务领域和电子消费品领域的商业项目经验。在后一领域,针对数码相机的会话式推荐系统已经得到了应用,它在一家奥地利的大型比价平台上被超过20万网上购物者使用。一个在线问卷的回复支持了这个假设,即导购应用帮助用户在面对大量选择时能更好地给自己定位。在使用这个会话式推荐系统时与没有使用的对比,明显更高比例的用户成功地完成了他们的购物。在理财服务领域,基于知识的推荐系统跟随着不同的商业模式,他们支持销售代理商,并与潜在用户相互作用。包括销售代表的经验主义调查指出,当与客户端交互时,节省的时间是非常大的优势,它反过来允许销售人员来鉴定销售机会[13, 17]。

[58]中的一个案例分析研究了一个基于知识的会话式销售推荐系统如何影响网上购物者的行为。他们分析了在引进推荐系统之前和之后两段时期的销售记录。这项研究的一个有趣发现是,在这两段时期内排名最高的物品列表非常不同。实际上,那些在前一阶段销量很少但是由系统推荐后却很高的物品,折射出了非常高的需求。因此,物品的相对销量能够明确地展现出与推荐系统推荐这些物品的频率的一一对应关系。由推荐应用给出的建议被用户采用并导致了在线转化。这也证实了最初讨论的像文献[15]中的用户研究结果。最后,另一个基于知识的旅游推荐系统的评估旨在对比转化率,也就是说,在交互式销售导购的用户与非交互式销售导购的用户之间,下单用户的比例[59]。这项研究有力地证明了,那些与交互式旅游向导发生交互的用户,发起一笔订单请求的概率是其他用户的两倍甚至更高。

因此,基于以上结果我们能够下结论,基于约束的推荐已经成功地应用在多处商业应用领域并得到这些应用用户的承认。

## 6.6 未来的研究方法

一方面,基于约束的推荐系统的效用已经在很多领域的应用中得到了证明;另一方面,对它的研究和应用仍然存在很多挑战和改进的空间。这些改进能帮助我们提升对用户的推荐质量,拓宽应用领域,并开发推荐软件。

**产品数据抽取自动化** 一个基于约束的推荐系统的好坏程度取决于它的知识库。所以,知识库一定要是正确的、完整的、最新的,以确保高的推荐质量。这些要求意味着巨大的维护工作,尤其在那些数据和推荐知识频繁改变的领域,如电子消费产品领域。现在,这些工作都由人类专家来完成,如收集产品数据或者更新基础规则。但是,在很多领域,至少那些产品数据能通过机器从互联网获取到的领域,通过互联网资源,很多推荐应

用需要的数据都可以获取收集得到。在这种情境下,主要的研究课题就是如何从不同的信息源来获取产品数据,并且自动检测和调整过时的数据。这些包含了识别相关数据源、抽取产品数据、解决矛盾数据问题等。一个最近的相关挑战来自于如何从书籍、CD、DVD和电视节目等数字多媒体中抽取产品数据。

但是,机器的问题在于如何表达互联网的数据,互联网的数据一般表示为人类易于获取和理解的信息的形式,遗憾的是,现在的机器却很难解释这些信息。因此,一个基础性的研究课题就在于我们如何使机器能够像人类一样能读懂这些信息。其实,这个问题已经超越推荐系统的研究范畴,它是语义网的一个核心问题,并且也是人工智能层面的一个需要解决的关键问题。尽管从现在来看,这个问题短期内还很难解决,但是我们仍然可以利用一些能应用到推荐中的数据。例如,在从网页中抽取产品数据的时候,我们可以以表格的形式来存储数据,并且去搜查数据,查找这些产品的描述,存储到产品数据库中[31]。当然,这种方式成功与否还需要看特定的领域。例如,在像数码相机这样的消费电子产品中,对相机的描述都遵循一个特定的结构(如不同品牌的相机的芯片都是相似的);而在另外一些领域,如度假产品,对产品大多数都是通过自然语言文本来描述的。需要提到的是,相对于将人类可读的语言翻译到机器可执行的数据,另一种方式就是将机器可执行的数据加到后面,或者直接替代人类才可读的内容。的确,在提供了机器可执行的数据的情况下,一些较强的市场力量(如搜索引擎)的服务会得到提升。例如,如果产品提供商提供了一些产品的特定格式的数据,那么他们的产品将会在搜索结果中获得更好的排序。但是在这种情境下,将这些用于搜索的描述应用到推荐意图中,仍然需要依赖于单一的授权问题。因此,让机器像人类一样读懂网页的工作仍然是一个重要的研究议题。

**基于社会化的知识获取** 基于约束的推荐的重要基础在于有效的知识获取和维护。过去,这个问题就在多个维度上被阐释,主要的关注点在于知识表达和问题的概念化,以及通过处理模型来获取和组织领域专家的知识。从历史来看,这些方法的一个重要假设就是存在一个知识的形式化方式,并且能导出一个面向用户的概念化和知识获取工具。在现实生活中的很多案例中,领域知识排在众多利益相关者的前头,典型的例子在于跨部门跨组织的商业规则或者新兴的应用中,在这里面,用户群体在开放创新的网络环境下,共享知识。最近,随着 Web 2.0 和语义网络技术的发展,协同获取知识的机会和问题又一次成为新的兴趣点。至于获取到的知识的类型,最近发展的关注重点在于获取结构化的知识,如在知识项的层面、知识概念层面,以及他们之间的关系等。现在新的方向是更进一步地尝试去协同地获取知识、提炼领域的约束、关注商业规则,因为在很多基于知识的应用中,它们是最重要的、最频繁变更的,因此也是最耗费人力物力的。我们主要需要回到以下几个问题:若知识是由不同的贡献者提供的,我们如何检测和修复冲突地问题?我们应该如何正确地提问,以让我们更好地从贡献者那里获取知识?我们应该如何提出好的提议,用以在不同的、可能只是部分定义的知识库中变更知识?

**术语知识获取**通常涉及支持使用特定语言的用户形式化规则、约束或者逻辑描述。这个任务在推荐系统中通常很复杂,因为推荐的输出包含了用户对推荐物品的喜好。结果就是,知识获取必须要支持形式化、调试、测试这些喜好的描述[21]。

更进一步,高质量的可解释的需求使得满足搜寻知识库的任务更复杂。在基于约束的推荐系统中,推荐理由的解释是利用知识库的内容生成的。事实上,不同的知识库可能会在相同输入的情况下,产生相同的推荐结果输出,但是推荐的解释却是不同的。因为一个更长远的重要的目标就在于让知识获取能够支持形式化可理解的知识库。

推荐系统的知识库是动态的,遗憾的是,这些动态不仅来自于产品分类的改变,同时也来自于用户喜好的改变。例如,在以前,将像素的数码照片打印在 A4 的纸上就能使用户满意,但是随着时间的推移,用户对质量的要求也变高了。因此自动的检测这种变化并且随之调整的知识库也是一个有趣的研究课题。

**验证** 成功开发和维护一个推荐知识库需要一个智能的测试环境,以确保推荐的正确性。特别是在一些对推荐质量有高要求的应用中(如理财产品),一个公司使用一个推荐系统,需要确保它的推荐过程和结果的质量。于是,未来的研究需要关注如何开发一种机制,能够自动配置测试环境,使得我们可以用最小的测试样例来尽可能地识别错误。最小化测试集是很重要的,因为领域专家需要人工来验证它们。这种验证输出与知识获取恰巧匹配,因为任何知识工程师的反馈都能为推荐知识库所用。特别地,一个有趣的研究问题在于如何利用用户喜欢或者讨厌的反馈来提升知识库。文献[51]介绍了一种算法来告诉我们如何研究知识库来回答为什么一个产品该推荐或者不推荐。

**组合产品和服务的推荐** 100 多年前,随着 Model T 的生产, Henry Ford 用大规模制造(许多同一商品的高效量产)颠覆了制造业。然而今天,大规模制造已是一个过时的商业模式,公司必须提供能够满足用户个性化需求的产品和服务。在这种环境下,大规模定制——在大规模制造的成本下,高度多样性产品和服务的量产——已成为新的商业典范。伴随着大规模定制,大规模的混乱也随之而来,面对海量的商品,用户难以选择。开发应用于结构化产品和服务的推荐技术有利于解决这种大规模混乱的难题。例如,推荐技术能够帮助不知情用户在一个几乎有无限产品的领域中发掘他的愿望、需求和对产品的要求。但是,结构化产品的推荐仍然受到现在技术手段的限制。当前的技术假设待推荐的物品能够具体地展示。但是,结构化领域常常提供这样的—一个高度产品差异,即一系列可能的组合集只能以结构化描述为内在特征。例如,结构化的系统或许包含了数以千计的部件和连接器。在这些领域,找到满足用户需求的最好组合是一个很有挑战性的问题。

**可理解性和可接受性** 为了具有说服力,推荐系统必须给出用户推荐理由。当用户可以质疑一次推荐,并且探究系统推荐一个特定的物品的原因时,用户就会开始相信这个系统。一般来说,推荐的理由(或解释)往往在给出推荐结果的时候一并给出,用以达到增加用户传递和信任、说服用户、提升用户满意度等目的。这些解释依赖推荐过程中的状态和用户的属性,例如,她的目标、需求、先验知识等。未来推荐系统的愿景是,对于推荐的解释也能够得到优化,例如,如果推荐系统识别到一个用户不理解推荐出来的不同产品的区别,那么这些区别就可以在推荐理由中得到说明。相反地,如果一个用户对产品完全了解,并且清晰地知道自己需要什么,那么推荐的解释则可以为他提供一个详细的技术判别。所以,未来的研究挑战在于如何创建一个人工推荐的机制,来灵活地满足不同用户的需求。在这些努力中,推荐理由和解释将是一个重要的基础。

**用户购买行为理论** 一个真正智能的推荐系统能够适应用户。这意味着,推荐系统能对用户建模,并且能预测用户在接受一些信息之后的反应。特别地,如果我们有一个对用户购买行为影响因素建模的模型,能够对推荐系统的下一步动作作出指导,那么对推荐系统的研究将很大程度上受益于认知和决策的理论。有人也许会从伦理道德的角度来对这种推荐系统提出质疑,但是用户提供的所有信息都会影响他的购买行为。为了更好地规划好推荐系统的开发,和用户进行沟通交流是一个很重要的工作。

**上下文和情景智能感知** 推荐系统不仅是一个简单的 PC 工具软件,而是在很多情况

下能做出推荐动作的智能系统。例如,在未来的辅助驾驶技术中,系统能够对各种动作提供建议,如超车、转弯、停车等。为了在这种情况下能提供推荐,推荐系统必须能够感知当时的情况,并且了解用户的目标。其他典型的情景包括假期度假的推荐,在这种情况下,推荐系统不仅需要知道用户的喜好,并且能够知道时间、季节、天气状况、票务信息等。注意,上文提到的情境的需求叫作环境智能。传统的电脑只是一个单一的和用户交互的界面接口,而语音、手势等都在用户和推荐系统的交互中扮演着重要的角色。

**语义网** W3C认为语义网提供了一个通用的平台,在这个平台上,数据能够跨越应用、企业和社区的界限,并被共享和复用。在特定的语义网里,技术能将数据联系起来,这可以用于实现一个分散的网络,人与人之间的信任关系、用户与商品的关系都可以在这个网络中表示。基于这种用户和商品的关系,很多应用都能得到提升。我们已经提到,可以从机器可读的信息中抽取产品信息和获取知识,其实我们可以更进一步,利用语义网的信息来帮助提升推荐的质量[22, 63]。尤其是,可以开发一种机制来识别有用的评分,从而避开一些人为的对系统的误导。语义网也允许我们在推理阶段整合更多的数据。一方面,由于知识是由分散网络社区中的很多人一起贡献和维护的,知识获取的任务也由大家一起完成,这有利于基于知识的推荐系统的提升。另一方面,很多的研究问题随之而来:如何保证推荐的质量?如何获取有用的和高质量的知识源?如果确保用到的产品和服务的描述在概念和价值层面达到大家的共识?如何同时保证推荐的正确性和完整性?

## 6.7 总结

本章大致介绍了基于约束的推荐系统技术,这些技术在协同过滤和基于内容的技术有明显缺点的情况下十分有用,并且能够应用到大型的复杂的产品类别系统中。基于约束的推荐系统的可用性已经在一些商业应用系统中得到证明,这些应用已经在本章进行了介绍和分析。最后,为了引出这个领域未来的研究课题,我们介绍了一些未来扩展的重要方向。

## 参考文献

1. Bistarelli, S., Montanary, U., Rossi, F.: Semiring-based Constraint Satisfaction and Optimization. *Journal of the ACM* **44**, 201–236 (1997)
2. Bridge, D.: Towards Conversational Recommender Systems: a Dialogue Grammar Approach. In: D.W. Aha (ed.) *Proceedings of the EWCBR-02 Workshop on Mixed Initiative CBR*, pp. 9–22 (2002)
3. Bridge, D., Goeker, M., McGinty, L., Smyth, B.: Case-based recommender systems. *Knowledge Engineering Review* **20**(3), 315–320 (2005)
4. Burke, R.: Knowledge-Based Recommender Systems. *Encyclopedia of Library and Information Science* **69**(32), 180–200 (2000)
5. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* **12**(4), 331–370 (2002)
6. Burke, R., Hammond, K., Young, B.: Knowledge-based navigation of complex information spaces. In: *Proceedings of the 13th National Conference on Artificial Intelligence, AAAI'96*, pp. 462–468. AAAI Press (1996)
7. Burke, R., Hammond, K., Young, B.: The FindMe Approach to Assisted Browsing. *IEEE Intelligent Systems* **12**(4), 32–40 (1997)
8. Burnett, M.: HCI research regarding end-user requirement specification: a tutorial. *Knowledge-based Systems* **16**, 341–349 (2003)
9. Chen, L., Pu, P.: Evaluating Critiquing-based Recommender Agents. In: *Proceedings of the 21st National Conference on Artificial Intelligence and the Eighteenth Innovative Ap-*



- plications of Artificial Intelligence Conference, AAAI/IAAI'06, pp. 157–162. AAAI Press, Boston, Massachusetts, USA (2006)
10. Felfernig, A.: Reducing Development and Maintenance Efforts for Web-based Recommender Applications. *Web Engineering and Technology* 3(3), 329–351 (2007)
  11. Felfernig, A., Burke, R.: Constraint-based recommender systems: technologies and research issues. In: *Proceedings of the 10th International Conference on Electronic Commerce, ICEC'08*, pp. 1–10. ACM, New York, NY, USA (2008)
  12. Felfernig, A., Friedrich, G., Jannach, D., Stumptner, M.: Consistency-based diagnosis of configuration knowledge bases. *Artificial Intelligence* 152(2), 213–234 (2004)
  13. Felfernig, A., Friedrich, G., Jannach, D., Zanker, M.: An integrated environment for the development of knowledge-based recommender applications. *International Journal of Electronic Commerce* 11(2), 11–34 (2007)
  14. Felfernig, A., Friedrich, G., Schubert, M., Mandl, M., Mairitsch, M., Teppan, E.: Plausible Repairs for Inconsistent Requirements. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI'09*, pp. 791–796. Pasadena, CA, USA (2009)
  15. Felfernig, A., Gula, B.: An Empirical Study on Consumer Behavior in the Interaction with Knowledge-based Recommender Applications. In: *Proceedings of the 8th IEEE International Conference on E-Commerce Technology (CEC 2006) / Third IEEE International Conference on Enterprise Computing, E-Commerce and E-Services (EEE 2006)*, p. 37 (2006)
  16. Felfernig, A., Isak, K., Kruggel, T.: Testing Knowledge-based Recommender Systems. *OE-GAI Journal* 4, 12–18 (2007)
  17. Felfernig, A., Isak, K., Szabo, K., Zachar, P.: The VITA Financial Services Sales Support Environment. In: *Proceedings of the 22nd AAAI Conference on Artificial Intelligence and the 19th Conference on Innovative Applications of Artificial Intelligence, AAAI/IAAI'07*, pp. 1692–1699. Vancouver, Canada (2007)
  18. Felfernig, A., Kiener, A.: Knowledge-based Interactive Selling of Financial Services using FSAdvisor. In: *Proceedings of the 20th National Conference on Artificial Intelligence and the 17th Innovative Applications of Artificial Intelligence Conference, AAAI/IAAI'05*, pp. 1475–1482. AAAI Press, Pittsburgh, PA (2005)
  19. Felfernig, A., Mairitsch, M., Mandl, M., Schubert, M., Teppan, E.: Utility-based Repair of Inconsistent Requirements. In: *Proceedings of the 22nd International Conference on Industrial, Engineering and Other Applications of Applied Intelligence Systems, IEAAIE 2009, Springer Lecture Notes on Artificial Intelligence*, pp. 162–171. Springer, Taiwan (2009)
  20. Felfernig, A., Shchekotykhin, K.: Debugging user interface descriptions of knowledge-based recommender applications. In: *Proceedings of the 11th International Conference on Intelligent User Interfaces, IUI 2006*, pp. 234–241. ACM Press, New York, NY, USA (2006)
  21. Felfernig, A., Teppan, E., Friedrich, G., Isak, K.: Intelligent debugging and repair of utility constraint sets in knowledge-based recommender applications. In: *Proceedings of the ACM International Conference on Intelligent User Interfaces, IUI 2008*, pp. 217–226 (2008)
  22. Gil, Y., Motta, E., Benjamins, V., Musen, M. (eds.): *The Semantic Web - ISWC 2005, 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6–10, 2005, Lecture Notes in Computer Science*, vol. 3729. Springer (2005)
  23. Godfrey, P.: Minimization in Cooperative Response to Failing Database Queries. *International Journal of Cooperative Information Systems* 6(2), 95–149 (1997)
  24. Herlocker, J., Konstan, J., Terveen, L., Riedl, J.: Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems* 22(1), 5–53 (2004)
  25. Jannach, D.: Advisor Suite - A knowledge-based sales advisory system. In: R.L. de Mantaras, L. Saitta (eds.) *Proceedings of European Conference on Artificial Intelligence, ECAI 2004*, pp. 720–724. IOS Press, Valencia, Spain (2004)
  26. Jannach, D.: Techniques for Fast Query Relaxation in Content-based Recommender Systems. In: C. Freksa, M. Kohlhase, K. Schill (eds.) *Proceedings of the 29th German Conference on AI, KI 2006*, pp. 49–63. Springer LNAI 4314, Bremen, Germany (2006)
  27. Jannach, D.: Fast computation of query relaxations for knowledge-based recommenders. *AI Communications* 22(4), 235–248 (2009)
  28. Jannach, D., Bundgaard-Joergensen, U.: SAT: A Web-Based Interactive Advisor For Investor-Ready Business Plans. In: *Proceedings of International Conference on e-Business*, pp. 99–106 (2007)
  29. Jannach, D., Kreutler, G.: Personalized User Preference Elicitation for e-Services. In: *Proceedings of the IEEE International Conference on e-Technology, e-Commerce, and e-Services, EEE 2005*, pp. 604–611. IEEE Computer Society, Hong Kong (2005)
  30. Jannach, D., Kreutler, G.: Rapid Development Of Knowledge-Based Conversational Recommender Applications With Advisor Suite. *Journal of Web Engineering* 6, 165–192 (2007)



31. Jannach, D., Shchekotykhin, K., Friedrich, G.: Automated Ontology Instantiation from Tabular Web Sources - The AllRight System. *Journal of Web Semantics* 7(3), 136–153 (2009)
32. Jannach, D., Zanker, M., Fuchs, M.: Constraint-based recommendation in tourism: A multi-perspective case study. *Information Technology and Tourism* 11(2), 139–156 (2009)
33. Junker, U.: QUICKXPLAIN: Preferred Explanations and Relaxations for Over-Constrained Problems. In: *Proceedings of National Conference on Artificial Intelligence, AAAI'04*, pp. 167–172. AAAI Press, San Jose (2004)
34. Konstan, J., Miller, N., Maltz, D., Herlocker, J., Gordon, R., Riedl, J.: GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM* 40(3), 77–87 (1997)
35. Lakshmanan, L., Leone, N., Ross, R., Subrahmanian, V.: ProbView: A Flexible Probabilistic Database System. *ACM Transactions on Database Systems* 22(3), 419–469 (1997)
36. Lorenzi, F., Ricci, F., Tostes, R., Brasil, R.: Case-based recommender systems: A unifying view. In: *Intelligent Techniques in Web Personalisation*, no. 3169 in *Lecture Notes in Computer Science*, pp. 89–113. Springer (2005)
37. Mahmood, T., Ricci, F.: Learning and adaptivity in interactive recommender systems. In: *Proceedings of the 9th International Conference on Electronic Commerce, ICEC'07*, pp. 75–84. ACM Press, New York, NY, USA (2007)
38. Maimon, O., Rokach, L. Data Mining by Attribute Decomposition with semiconductors manufacturing case study, in *Data Mining for Design and Manufacturing: Methods and Applications*, D. Braha (ed.), Kluwer Academic Publishers, pp. 311–336 (2001)
39. McSherry, D.: Incremental Relaxation of Unsuccessful Queries. In: P. Funk, P.G. Calero (eds.) *Proceedings of the European Conference on Case-based Reasoning, ECCBR 2004*, no. 3155 in *Lecture Notes in Artificial Intelligence*, pp. 331–345. Springer (2004)
40. McSherry, D.: Retrieval Failure and Recovery in Recommender Systems. *Artificial Intelligence Review* 24(3-4), 319–338 (2005)
41. Mirzadeh, N., Ricci, F., Bansal, M.: Feature Selection Methods for Conversational Recommender Systems. In: *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service on e-Technology, e-Commerce and e-Service, EEE 2005*, pp. 772–777. IEEE Computer Society, Washington, DC, USA (2005)
42. Pazzani, M.: A Framework for Collaborative, Content-Based and Demographic Filtering. *Artificial Intelligence Review* 13(5-6), 393–408 (1999)
43. Peischl, B., Nica, M., Zanker, M., Schmid, W.: Recommending effort estimation methods for software project management. In: *Proceedings of the International Conference on Web Intelligence and Intelligent Agent Technology - WPRRS Workshop*, vol. 3, pp. 77–80. Milano, Italy (2009)
44. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic Critiquing. In: *Proceedings of the 7th European Conference on Case-based Reasoning, ECCBR 2004*, pp. 763–777. Madrid, Spain (2004)
45. Reiter, R.: A theory of diagnosis from first principles. *Artificial Intelligence* 32(1), 57–95 (1987)
46. R.Elmasri, Navathe, S.: *Fundamentals of Database Systems*. Addison Wesley (2006)
47. Ricci, F., Mirzadeh, N., Bansal, M.: Supporting User Query Relaxation in a Recommender System. In: *Proceedings of the 5th International Conference in E-Commerce and Web-Technologies, EC-Web 2004*, pp. 31–40. Zaragoza, Spain (2004)
48. Ricci, F., Mirzadeh, N., Venturini, A.: Intelligent query management in a mediator architecture. In: *Proceedings of the 1st International IEEE Symposium on Intelligent Systems*, vol. 1, pp. 221–226. Varna, Bulgaria (2002)
49. Ricci, F., Nguyen, Q.: Acquiring and Revising Preferences in a Critique-Based Mobile Recommender System. *IEEE Intelligent Systems* 22(3), 22–29 (2007)
50. Ricci, F., Venturini, A., Cavada, D., Mirzadeh, N., Blaas, D., Nones, M.: Product Recommendation with Interactive Query Management and Twofold Similarity. In: *Proceedings of the 5th International Conference on Case-Based Reasoning*, pp. 479–493. Trondheim, Norway (2003)
51. Shchekotykhin, K., Friedrich, G.: Argumentation based constraint acquisition. In: *Proceedings of the IEEE International Conference on Data Mining* (2009)
52. Smyth, B., McGinty, L., Reilly, J., McCarthy, K.: Compound Critiques for Conversational Recommender Systems. In: *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, WI'04*, pp. 145–151. Maebashi, Japan (2004)
53. Thompson, C., Goeker, M., Langley, P.: A Personalized System for Conversational Recommendations. *Journal of Artificial Intelligence Research* 21, 393–428 (2004)
54. Tsang, E.: *Foundations of Constraint Satisfaction*. Academic Press, London and San Diego (1993)

55. Williams, M., Tou, F.: RABBIT: An interface for database access. In: Proceedings of the ACM '82 Conference, ACM'82, pp. 83–87. ACM, New York, NY, USA (1982)
56. Winterfeldt, D., Edwards, W.: Decision Analysis and Behavioral Research. Cambridge University Press (1986)
57. Zanker, M.: A Collaborative Constraint-Based Meta-Level Recommender. In: Proceedings of the 2nd ACM International Conference on Recommender Systems, RecSys 2008, pp. 139–146. ACM Press, Lausanne, Switzerland (2008)
58. Zanker, M., Bricman, M., Gordea, S., Jannach, D., Jessenitschnig, M.: Persuasive online-selling in quality & taste domains. In: Proceedings of the 7th International Conference on Electronic Commerce and Web Technologies, EC-Web 2006, pp. 51–60. Springer, Krakow, Poland (2006)
59. Zanker, M., Fuchs, M., Höpken, W., Tuta, M., Müller, N.: Evaluating Recommender Systems in Tourism - A Case Study from Austria. In: Proceedings of the International Conference on Information and Communication Technologies in Tourism, ENTER 2008, pp. 24–34 (2008)
60. Zanker, M., Jessenitschnig, M.: Case-studies on exploiting explicit customer requirements in recommender systems. User Modeling and User-Adapted Interaction: The Journal of Personalization Research, A. Tuzhilin and B. Mobasher (Eds.): Special issue on Data Mining for Personalization 19(1-2), 133–166 (2009)
61. Zanker, M., Jessenitschnig, M., Jannach, D., Gordea, S.: Comparing recommendation strategies in a commercial context. IEEE Intelligent Systems 22(May/Jun), 69–73 (2007)
62. Zhang, J., Jones, N., Pu, P.: A visual interface for critiquing-based recommender systems. In: Proceedings of the 9th ACM Conference on Electronic Commerce, EC'08, pp. 230–239. ACM, New York, NY, USA (2008)
63. Ziegler, C.: Semantic Web Recommender Systems. In: Proceedings of the EDBT Workshop, EDBT'04, pp. 78–89 (2004)

# 情境感知推荐系统

Gediminas Adomavicius 和 Alexander Tuzhilin

**摘要** 在电子商务个性化、信息检索、便携移动计算、数据挖掘、营销和管理学等领域中,研究者和从业者已经开始意识到情境信息的重要性。虽然已就推荐系统领域开展了大量的研究,但是大多数已有的方法只是关注把最相关的物品推荐给用户,却忽略了相关情境信息,如时间、地点,或是否有人陪同(如看电影或外出就餐)。本章指出,情境信息在推荐系统中有一定的影响,是提供推荐方案时的重要参考。我们将讨论情境信息的一般概念,以及如何在推荐系统里面对情境信息建模。此外,我们将介绍三种不同的算法形式——基于情境的预过滤、后过滤以及建模——从而把情境信息纳入推荐流程中,讨论把几种把情境感知推荐技术整合为一个统一框架的可行性,并通过一个案例研究这种整合方法。最后,我们将讨论情境感知推荐引擎的其他作用,并讨论在将来的研究中重要以及有潜力的方向。

## 7.1 简介

大多数已有的方法只是把最相关的物品推荐给用户,而不考虑其他的环境因素,如时间、地点或是否有人陪同(如看电影或外出就餐)等。换句话说,传统的推荐系统只应用于两类实体,即用户和物品,而并没有将它们放入某种情境中进行考虑。

然而,在许多推荐系统的应用中,如推荐一个旅行套餐、个性化的网站内容或某部电影时,只考虑用户和物品可能是不够的——在某些特定的场景下给用户推荐物品,把情境信息整合到推荐流程里也是很有必要的。例如,在考虑了温度的因素后,旅游推荐系统在冬季推荐的度假地可能和夏天推荐的度假地大不相同。同理,在 Web 站点上提供个性化的内容的时候,我们也需要确定把什么内容在什么时候推荐给访客。比如,在工作日时,用户上午访问网站时可能会倾向于浏览环球新闻,晚上则会浏览股市报告;而在周末则会浏览影评和购物信息。

这些观察和消费者的决策行为经研究是一致的:消费者所做的决策与当时的情境是相关的。因此,在推荐系统里,情境信息整合到推荐方法中的广度和深度,毫无疑问会影响消费者偏好的预测准确度。

最近,很多公司已经开始把一些情境信息引入推荐引擎里。例如,互动式电台 Sourcetone ([www.sourcetone.com](http://www.sourcetone.com))在给用户挑选歌曲时,会考虑听众所给出的他当时的心情(情境)。在音乐推荐引擎中,一些情境信息,如听众的情绪,会对提供更好的推荐起到帮助。然而,我们还不确定情境信息是否会影响其他更多的推荐应用。

---

Gediminas Adomavicius, Department of Information and Decision Sciences Carlson School of Management, University of Minnesota e-mail: [gedas@umn.edu](mailto:gedas@umn.edu)

Alexander Tuzhilin, Department of Information, Operations and Management Sciences Stern School of Business, New York University e-mail: [atuzhili@stern.nyu.edu](mailto:atuzhili@stern.nyu.edu)

翻译:张杰 审核:郑重,洪弘,肖铨武,郑州大学-吴宾

本章讨论的课题是情境感知推荐系统(CARS)以及相关的一些其他问题,它试图说明在某些应用领域,在数据可用的前提下,情境信息确实有助于产生更好的推荐。我们会介绍三种主要的将情境信息整合进推荐系统里的方法,逐一分析这三种方法,以及如何将这三种方法统一。把情境信息引入推荐过程中,会给终端用户和推荐系统带来更丰富多样的互动。因此,在本章我们还会介绍能用于情境感知推荐系统的推荐查询语言(recommendation query language),并探讨其灵活的交互能力。

本章其余部分安排如下:7.2节讨论情境信息的一般概念以及如何在推荐系统里对情境信息建模。7.3节提出了把情境信息纳入推荐过程的三种不同的算法形式。7.4节讨论了把几种情境感知推荐技术整合成一个统一方法的可能性,并通过一个案例研究这种整合方法。情境感知推荐系统的一些其他重要功能将在7.5节讨论,7.6节会给出本章的结论以及在该领域未来的一些研究方向。

## 7.2 推荐系统中的情境

讨论情境信息在推荐系统的作用和带来的机遇前,在7.2.1节先讨论情境的基本概念。然后,在7.2.2节重点关注推荐系统,并解释情境是如何定义并在推荐系统中建模的。

### 7.2.1 什么是情境

情境是一个多方面的概念,在不同的学科,包括计算机科学(主要是在人工智能和普通计算、认知科学、语言学、哲学、心理学、组织科学等都进行了其相关研究。实际上,已经有一个完整的学术会议——CONTEXT(如 <http://context-07.ruc.dk>)——在专门研究这个话题,并把它与医学、法律、商学等其他学科相结合。例如,知名商学研究者和从业者C. K. Prahalad指出,“想具有随时随地能感知客户、了解客户需求的能力,意味着企业不仅要提供有竞争力的产品,也需要提供别具一格,根据客户所处环境而改变的实时客户体验”,这会是CRM从业者的下一个主要研究问题(“大事件”)[57]。

由于很多学科都对情境进行了研究,每门学科往往有其自己特有的观点,这些因素导致在不同学科中,“情境”(context)这个词的意义相对于字典的标准定义“一些能够对某些事情产生影响的条件和环境”要更具体一些[70]。因此,在不同的学科,甚至在这些学科的特定子领域,都存在着对情境的不同定义。

Bazire 和 Brézillon[17]找到并分析了不同领域的共计150种情境的定义。考虑到情境概念的复杂性和多面性,150种并不算夸张。Bazire 和 Brézillon[17]提到:

“……很难找到一个能适用于任何学科的情境的定义。情境是给定对象的一个框架吗?是能够对对象产生影响的所有因素的集合吗?是否可以把情境信息定义为影响对象的先验因素,还是仅将其理解为导致事件出现偏差的后验因素?它是静态的还是动态的?在人工智能领域出现了一些新的方法……在心理学中,我们通常在一个既定情况下去研究一个人的行为。哪些情境跟我们的研究有关?跟人有关的情境?跟任务有关的?跟交互有关的?跟当时形势有关的?情境什么时候开始,什么时候停止呢?情境和认知之间的真实关系又是什么呢?”

由于本书专注于推荐系统,也因为情境的概念非常宽泛,我们试图把重点放在与推荐系统有直接关系的领域,如数据挖掘、个性化电子商务、数据库、信息检索、移动情境感知系

统、市场营销和管理学等。在这些领域中我们遵循 Palmisano 等[54]对“情境”概念的描述。

**数据挖掘。**在数据挖掘中,情境有时被定义为能够标识顾客生活阶段(life stage)的特征性事件,这些事件可以改变他/她的喜好、状态和商业价值[18]。例如,人的特定生活阶段:新工作、孩子出生、结婚、离婚、退休等。了解情境信息有利于:(a)只选择相关的数据来挖掘针对某个情境的模式,例如,跟女儿的婚礼有关的数据;(b)只选择相关的结果,即选择那些适用于特定情境的数据挖掘结果,如已发现某人退休的模式。

**个性化电子商务。**Palmisano 等[54]在电子商务场景中,使用客户的购买意向作为一个情境。不同的购买意向可能会导致不同的购买行为。例如,同一个账号下同一个客户会针对不同的原因购买不同的产品:买一本书来提高个人的技能,买一本书作为礼物,或因爱好而购买电子设备。为了分析不同的采购意向,Palmisano 等[54]为每一个客户针对每一次购买的情境建立单独的档案,并利用这些档案建立独立的顾客模型。对某些顾客细分群,在特定的情境下运用建立的顾客模型来预测顾客的购买行为。这种依赖于情境的客户细分群十分有用,因为它能更好地预测顾客在不同的电子商务业务的购买行为[54]。

因为要向客户提供个性化的产品和服务推荐,所以推荐系统也与个性化电子商务有关。在推荐系统中纳入并使用情境信息重要性已在文献[3]中证明,作者提出了一个多维的方法,除了使用传统的用户和物品信息进行推荐,也可以基于情境信息提供推荐。Adomavici 等[3]也说明了情境信息对推荐系统的意义:在某些情况下,它有助于提高推荐的质量。

同样,Oku 等[53]在推荐过程中加入额外的情境维度(如时间、同伴和天气),并使用机器学习技术在餐厅推荐系统提供推荐。经验表明,情境感知的方法在推荐精度和用户满意度明显优于同类的非情境感知方法。

由于本书重点讨论情境在推荐系统的应用,所以将在后面的章节描述这类方法。

**普适移动情境感知系统。**在移动情境感知系统文献中,情境最初定义为用户的地理位置、用户身边的人群的身份及用户附近的物品,还有这些因素的变化[63]。其他因素也陆续被添加进来。例如,Brown 等[23]把日期、季节和温度纳入考虑。Ryan 等[61]增加了对用户的身体和精神状态的研究。Dey 等[33]则把用户的情绪状态也纳入情境,同时将情境的概念拓展为“任何可以描述,且和用户与业务交互有关的信息”。有些研究认为情境信息与用户关联[33, 35],而另一些则强调情境与业务相关[60, 69]。最近,出现了一些其他的可用于情境感知系统的技术,包括在移动应用中采用的混合技术[59, 71]、视觉推荐系统中的图形化模型等[20]。

情境信息对给移动用户提供的各种基于位置的服务(LBSes)是至关重要的[64]。例如,百老汇剧院可能想在演出开始前 30min,给纽约时代广场游客推荐打折力度很大的剧院门票(因为这些门票在演出开始后就会浪费),并把这些信息发送到游客的智能电话或其他通信装置里。在这个业务场景,位置和通信终端的类型(如智能电话)构成本次业务的情境信息。Brown 等[22]给出了另一个有趣的业务应用,允许游客与远方的朋友分享他们的观光体验,这体现了情境感知技术在社交活动中的价值。

文献[30]是一份关于情境感知移动计算的研究调查,其中讨论了情境信息的各种模型、情境感知技术、不同的业务架构以及一些情境感知的应用实例。

**数据库。**某些数据库管理系统已经可以支持情境查询功能。在这里,查询语句综合考虑了用户偏好,将返回不同的查询结果。查询结果依赖查询语句在何种情境中被解析,以及与该情境下的用户偏好。例如,Stephanidis 等[66]引入了一组情境参数,同时为每个常规的关系属性集(即原有的数据库属性)和情境参数(即新增加的情境属性)的组合定义了一个偏好



(偏好是为每个组合定义的)。然后他们提出了一种情境感知的 SQL 语句扩展,以适应偏好和情境。Agrawal 等[7]提供了另一种方法整合情境和用户喜好的查询语言,通过开发对各种偏好进行综合评分的方法,能迅速给情境查询提供评分后的结果。Mokbel 和 Levandoski 的[52]描述了能够支持情境感知和位置感知的数据库服务器 CoreDB,并讨论了实现该系统的几个问题,包括情境感知的查询操作符、连续查询、多目标查询处理、查询优化等。

**信息检索。**情境信息已被证明对信息检索和访问有帮助[40]。然而大多数现有的系统检索决策仅仅基于查询语句和文档集合,而搜索的情境信息往往会被忽视[9]。主动检索系统的效果取决于执行基于情境的检索的能力,查询返回情境相关的结果[46, 65]。在 Web 搜索中,情境被视为与搜索关键字潜在相关的主题集合。Lawrence[45]介绍了如何使用情境信息,并提出了一些针对不同领域的基于情境的搜索引擎。Maamar 等[51]建议把情境信息集成到 Web 服务中。当前大多数情境感知的信息服务和检索技术考虑的是短期问题和用户即时的兴趣和需求(如找到一个春季会议期间创建的所有文件,找到的结果是“当时是阳光明媚的一天,在纽约一家意大利餐厅外面”),而不是对用户的长期兴趣和偏好去设计和建模。

**营销和管理。**市场研究人员已经证实购买过程在很大程度上取决于购买时的情境,因为同样的客户会根据情境做出不同的决策和选择不同的产品或品牌[19, 50]。根据 Lilien 等的结论[47],“消费者的决策规则会受到商品的使用场景,商品或服务的用途(送给家人、礼物或自用),购买的场景(广告目录的销售、商店内货架的选购、销售人员辅助购买)的影响而变化”。因此,能否准确预测消费者的购买偏好,取决于我们在多大程度将相关的情境信息整合进来。在市场研究领域,情境也是行为决策理论领域的研究方向。在 Lussier 和 Olshavsky[50]中,情境被定义为品牌选择策略中任务的复杂性。

Prahalad 定义情境是[57]“在给定的任何时间,客户确切的物理位置,那一秒钟,他/她所需要的服务,以及他/她的能够享受服务体验的移动设备”。此外, Prahalad[57]关注那些能够利用情境信息提供“独特的、实时的客户体验”的移动应用,而非提供有竞争力的产品。Prahalad[57]提供了一个例子,有一次他把笔记本电脑落在波士顿一家酒店,他很愿意付高额的费用让酒店把电脑邮寄给他(他当时在纽约且急需使用那台电脑)。

概括地说, Prahalad[57]把情境信息区分为三个维度:时间(提供客户体验的时间)、空间(提供推荐的地点)、技术(通过何种方式去提供推荐)。Prahalad 虽然侧重于实时的经验(这意味是目前的时间,即“现在”),但是时间维亦可以推广到过去和未来(例如,我想在明晚看一场电影)。

这一节的内容清楚地表明,情境是一个多方面的概念,适用于多种学科,每个学科都有自己独特的视角,并为这一概念打上自己的印记。所以要给这些多样性的观点带来一些“秩序”,由此 Dourish[34]引入了情境分类的概念,进而将各种对情境的定义归纳为“表征性”和“交互性”两种观点。表征性观点认为情境是一组预定义可观察的属性,其结构(或模式(schema),用数据库术语)不随时间的推移显著改变。换句话说,表征性的观点认为情境属性是可识别的,并且先验已知的,因此,可以获得并在情境感知业务中应用。与此相反,交互性的观点认为用户行为是相关的情境中引发的,情境本身是不可观察的。此外, Dourish[34]假设不同类型的行动会产生和要求不同类型的情境信息,基于此可以得到活动和相关的情境之间的双向关系:情境影响活动,而不同的活动也导致了不同的情境。

在 7.2.2 节中,我们将考虑这些针对情境的不同的定义和方法,并使他们适应推荐系统需求本身的特质。因此,我们也将修订和改进之前在文献[3, 53, 72]等中对推荐系统所用情境的定义。

### 7.2.2 在推荐系统实现情境信息的建模

推荐系统成为一个独立的研究领域是在 20 世纪 90 年代中期,研究人员和从业者开始关注推荐问题,当时依赖于显式的评分来获取用户对不同的物品的喜好度。例如,对于电影推荐系统,John Doe 可能给“Gladiator”评 7 分(满分 10 分),即  $R_{movie}(\text{John Doe}, \text{Gladiator})=7$ 。推荐的流程通常是从已有的一套评分集开始的,这些评分由用户提供,或由系统隐式推断。当这些初始评分数据准备好后,推荐系统会推导出评分函数  $R$  给尚未评分的  $(\text{user}, \text{item})$  对进行评分:

$$R: \text{User} \times \text{Item} \rightarrow \text{Rating}$$

评分是完全有序的(例如,在一定范围内的非负整数或实数),User 和 Item 分别是用户和物品的取值空间。一旦得出整个  $\text{User} \times \text{Item}$  空间的  $R$  函数,推荐系统可以为每个用户推荐评分最高的物品(或前  $k$  个最高评分的物品)。称为传统或二维(2D)系统,因为他们在推荐流程里面只考虑用户和物品两个维度。

换句话说,最普遍的做法是,推荐问题可以归纳为预测一个用户未接触过的物品的评分。这个预测通常是基于该用户对其他物品的评分、其他用户对该物品的评分以及一些其他可以利用的信息(如用户的自然属性、物品特征)给出的。值得注意的是,尽管现有的推荐系统研究已经在该领域进行了大量的工作,但绝大多数现有的方法在推荐物品给用户或推荐用户给物品时没有考虑额外的情境信息,如时间、地点或同伴(如看电影)。基于此,本章探索情境感知推荐系统(CARS),该系统通过将已有的情境信息作为附加的数据类型整合到推荐流程里,来处理对用户的兴趣和偏好的建模和预测问题。这种长期的偏好和兴趣通常以评分来标识,其对应的模型不仅是物品和用户的函数,也是情境信息的函数。换句话说,评分函数定义为

$$R: \text{User} \times \text{Item} \times \text{Context} \rightarrow \text{Rating}$$

其中,User 和 Item 分别是用户和物品的取值域,Rating 是评分的取值域,Context 声明了与应用相关的情境信息。为了进一步澄清这些概念,请看下面的例子。

**例 7.1** 在把电影推荐给用户的应用里,用户和电影的关系描述包含以下的属性:

- 电影:所有的可以被推荐的电影的集合,它被定义为  $\text{Movie}(\text{MovieID}, \text{标题}, \text{长度}, \text{发行年份}, \text{导演}, \text{类型})$ 。
- 用户:获得电影推荐的用户,它被定义为  $\text{User}(\text{UserID}, \text{姓名}, \text{住址}, \text{年龄}, \text{性别}, \text{职业})$ 。

另外,情境信息也包括了以下三种类型的属性信息:

- 剧院:电影的电影院,它被定义为  $\text{Theater}(\text{TheaterID}, \text{名称}, \text{地址}, \text{容量}, \text{城市}, \text{省}, \text{国家})$ 。
- 时间:电影放映的时间,它被定义为  $\text{Time}(\text{日期}, \text{DayOfWeek}, \text{TimeOfWeek}, \text{月}, \text{季度}, \text{年})$ 。在这里,属性 DayOfWeek 指周一、周二、周三、周四、周五、周六、周日,属性 TimeOfWeek 指“工作日”和“周末”。
- 伙伴:代表了跟该用户一起去看电影的一个人或一组人。它被定义为  $\text{Companion}(\text{伙伴类型})$ ,其中的属性伙伴类型有“独自”“朋友”“男/女朋友”“家庭”“同事”和“其他”。

用户对一部电影的评分也跟用户看电影时的位置、伙伴以及时间有关系。例如,推荐给大学生 Jane Doe 看的电影可能是明显不同的类型,这取决于她是打算星期六晚上跟男朋友去看,还是平日跟父母去看。

从这个例子和别的案例来看，我们可以看到情境信息里的 Context 可以有不同类型，每种类型定义了情境的某一方面，如时间、地点（如剧院）、伙伴（例如，去看电影的伙伴）或购买目的等。另外每个情境的类型都可能是一个反映了情境信息复杂性的复杂数据结构。虽然这种情境信息的复杂度可以有不同的形式，一种比较流行的结构是树状的层次结构，这在大多数的情境感知推荐和资料系统比较常见，包括[3]和[54]。比如，有三种情境会有以下相关的层次关系：剧院：TheaterID→城市→省→国家；时间：日期→DayOf-Week→TimeOfWeek；日期→月→季度→年<sup>⊖</sup>。

此外，我们根据 Dourish[34]的表征性观点，在 7.2.1 节描述，将情境定义为可预先观察变量的集合，那么这些数据结构并不会随着时间变化而发生重大的改变。虽然也有一些论文在文献利用交互的方法给情境推荐建模，如文献[11]通过借用心理学的短期记忆 (STM)的相互作用的方法对情境建模，大部分情境感知推荐的研究遵循表征性的观点。如前所述，本章也会采用这种表征性的观点，我们会假设在一个给定的环境中，有一个预定义的情境类型的有限集合，每种类型都有一个明确的结构。

更具体地说，在一定程度上我们遵循 Palmisano 等[54]和 Adomavicius 等的看法[3]，本节用情境维度(contextual dimension)集合  $\mathbf{K}$  定义情境信息，每个  $\mathbf{K}$  中的情境维度  $K$  都包含  $q$  个属性  $K = (K^1, \dots, K^q)$ ，其中每个属性都代表一类情境特征，且有层级结构，如时间和通信设备。底层属性  $K^q$  的值反映了细粒度的情境特征，而顶层属性  $K^1$  的值反映了粗粒度的情境特征。例如，图 7.1a 给出了一个具有四层结构的情境属性  $K$  的例子，用以说明电商零售业务中的购买交易的意图。顶层(最粗级)结构定义了  $K$  在所有可能情境下的购买情况，下一级属性  $K^1 = \{\text{Personal}, \text{Gift}\}$ ，表示客户的购买意图：作为个人用途购买还是作为礼物购买，进而到更精细的层次，“个人”属性  $K^1$  的“Personal”值可以进一步划分成更详细的个人背景：个人用途购买是跟工作相关的还是其他。同样，礼品的值  $K^1$  可细分为送给合作伙伴或朋友还是送给父母或其他人。因此， $K^2$  层级可以定义为  $K^2 = \{\text{PersonalWork}, \text{PersonalOther}, \text{GiftPartner/Friend}, \text{Gift-Parent/Other}\}$ 。最后， $K^2$  属性还可以进一步划分为更精细的层级，如图 7.1a 所示<sup>⊖</sup>。

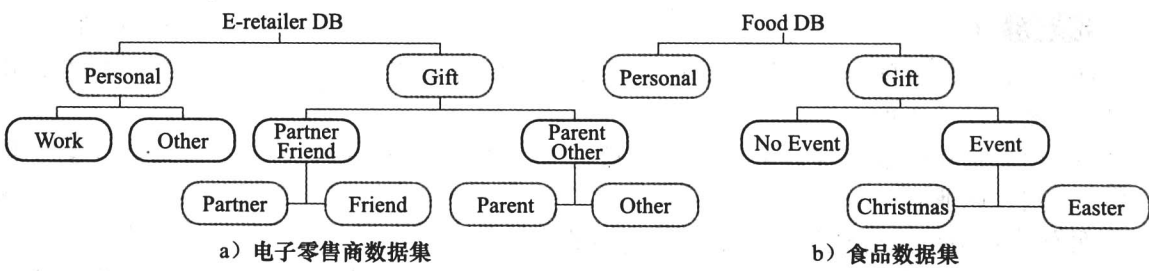


图 7.1 情境信息的层级结构

本小节介绍在文献[3]中给出的情境信息的定义：给传统的用户和物品的维度增加了情境信息，如时间、地理位置等，也可以通过基于 OLAP<sup>Ⓢ</sup>的多维数据模型(MD)引入，该模型在数据仓库应用中被广泛使用[29, 41]。形式上， $D_1, D_2, \dots, D_n$  代表维度，其中两个维

⊖ 为了保证完整性，需要指出的是，不仅在情景维度，传统的用户和物品维度也有层次关系的属性。比如，实例 1 的两个主要维度也可能有如下与之相关的层次：电影：电影编号→类型；用户：用户编号→年龄，用户编号→性别，用户编号→职业。

Ⓢ 出于简洁和示意的目的，这个图只做了两路划分。很明显，三路、四路乃至多路划分都是允许的。

Ⓢ OLAP 表示联机分析处理，这是一种操作和分析存储在多维立方结构中数据的常用方法，并广泛运用于决策支持系统。

度是用户和物品，其余维度代表情境信息。每个维度  $D_i$  是某些属性  $A_{ij}$ , ( $j=1, \dots, k_i$ ) (或称域) 的笛卡儿积的子集, 即  $D_i \subseteq A_{i1} \times A_{i2} \times \dots \times A_{ik_i}$ , 其中每个属性定义一个域 (或一组域) 的值 (即有自己的取值空间)。此外, 一个或多个属性可以形成一个键, 即它们唯一定义了属性的其余部分 [58]。在某些情况下, 一个维度可以由一个单一的属性定义, 这种情况下  $k_i = 1$ 。例如, 设想一下三维的推荐空间,  $\text{User} \times \text{Item} \times \text{Time}$ , 用户维度可以定义为  $\text{User} \subseteq \text{UName} \times \text{Address} \times \text{Income} \times \text{Age}$ , 即包含了具有姓名、地址、收入、年龄的属性集合。同样地, 物品维度可以定义为  $\text{Item} \subseteq \text{IName} \times \text{Type} \times \text{Price}$ , 即包含了具有物品名、类型和价格的属性集合。最后, 时间维度可以定义  $\text{Time} \subseteq \text{Year} \times \text{Month} \times \text{Day}$ , 即包含了从开始到结束时间的日期列表 (例如, 从 2003 年 1 月 1 日到 2003 年 12 月 31 日)。

给定  $D_1, D_2, \dots, D_n$  维度, 定义这些维度的空间笛卡儿积  $S = D_1 \times D_2 \times \dots \times D_n$ 。此外, 令  $\text{Rating}$  表示所有可能评分值的有序集合, 那么, 评分函数可被定义为空间  $D_1 \times \dots \times D_n$  上的函数

$$R: D_1 \times \dots \times D_n \rightarrow \text{Rating}$$

继续上面  $\text{User} \times \text{Item} \times \text{Time}$  的例子, 定义一个推荐空间  $\text{User} \times \text{Item} \times \text{Time}$  上的评分函数  $R$ , 用以说明用户  $u \in \text{User}$  在时间点  $t \in \text{Time}$  对物品  $\text{item} i \in \text{Item}$  的喜好程度。

直观上看, 定义在推荐空间  $S = D_1 \times D_2 \times \dots \times D_n$  上的评分函数  $R(d_1, \dots, d_n)$  可以表示为一个多维数据立方体, 如图 7.2 所示, 每个立方体块保存了推荐空间  $\text{User} \times \text{Item} \times \text{Time}$  的某个评分  $R(u, i, t)$ , 图中的三个表格分别定义了跟  $\text{User}$ 、 $\text{Item}$  和  $\text{Time}$  维度关联的用户、物品、时间索引 (即取值范围)。例如, 图 7.2 中的评分  $R(101, 7, 1) = 6$  中代表为  $\text{User ID}$  101 的用在在工作日期间对  $\text{Item ID}$  为 7 的物品评分为 6。

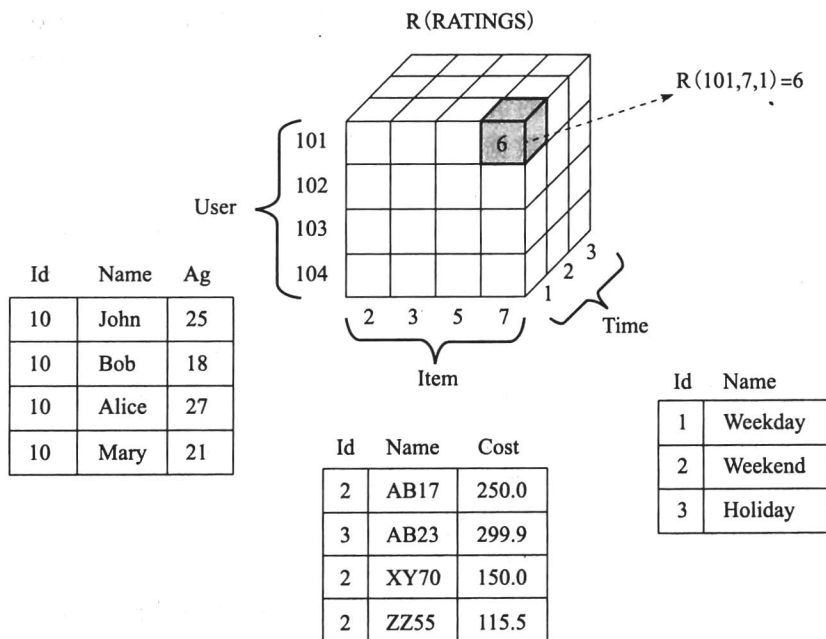


图 7.2 推荐空间  $\text{User} \times \text{Item} \times \text{Time}$  的多维模型

上述评分函数  $R$  通常是局部函数 (partial function, 即不是所有推荐空间中的点都有定义), 且已预先知道某些初始点上的评分值。和一般推荐系统一样,  $R$  的目标是估计剩下未知点的评分值, 并将评分函数  $R$  推广到全局。

上面描述的基于传统 OLAP 的多维情境模型 (以下简称 MD 模型) 和之前一般的情境

层级模型之间的主要差别在于,之前案例里的情境信息都是以一些更普遍的层级分类保存,如树(平衡和非平衡)、有向无环图(DAG)或者各种其他类型的分类法。另外,MD模型的评分保存在多维立方体中,而一般情境模型的评分则保存在更一般的层次结构中。

还想指出的是,并非所有的情境信息都有助于推荐,如书的推荐系统。推荐系统可以采集买书者的各种情境数据,包括:(a)买书的目的(选项是:为了工作,为了休闲,...), (b)计划的阅读时间(工作日,周末,...); (c)计划阅读的地点(在家,在学校,在飞机上,...), (d)买书时股市的指数。显然某些类型的情境信息在特定的场景中可能比其他类型的信息更重要。例如,前面的例子中,股票指数跟买书的意图比起来,相关性就显得不太重要。有几种方法可以衡量情境信息与推荐动作之间的相关性。相关性度量可以手动完成,例如,参考推荐系统设计者或领域专家的知识;或自动度量,比如,在数据预处理阶段采用现有的大量基于机器学习[42]、数据挖掘[48]和统计学[28]的特征选择方法。特征选择方法的详细讨论超出了本书的范围,在本章的剩余部分,我们会假定数据只存储了相关的情境信息。

### 7.2.3 获取情境信息

情境信息可以通过许多方法获得,包括:

- **显式获得**,即通过直接接触相关人士和其他情境信息源,通过直接问问题或者引导性的方式显式获取这些信息。例如,一个网站获取情境信息会让用户填写某些调查表,等用户回答完相关问题后才能访问某些页面。
- **隐式获得**,隐式地从数据或环境中获得,例如,从移动电话公司获得用户的位置变动信息。又如,可以从一个事务的时间戳隐式地获得时间的情境信息。我们不需要跟用户或其他情境信息源进行直接的交互,就可以直接从隐式的情境信息源中直接访问并提取我们需要的情境信息。
- **推断获得**,我们通过统计和数据挖掘方法推断出情境信息。例如,有线电视公司可能很难显式获得电视观众的家庭角色(丈夫、妻子、儿子、女儿等),但可以通过所观看的电视节目相对合理和准确地推断出来。为了推断情景信息,有必要创建一个预测模型(例如,分类器)并使用合适的数据训练该模型。能否成功推断该情景信息很大程度上取决于这样的分类器,并且在不同的应用中有相当大的差异。如文献[54]中描述,在特定应用中使用特定数据挖掘方法可以推断出准确度相当高的不同类型的情景信息,例如朴素贝叶斯分类器和贝叶斯网络。

最后,情境信息有时会以一种潜在的形式“隐藏”在数据里,我们并不一定要显式地得到情境信息,可以直接隐式地使用这些信息以获得对未知情形更精确的评分。例如,在前面的例子中,我们把家庭角色(丈夫、妻子等)对电视节目的喜好倾向作为潜在变量进行建模来预测一个人对某电视节目的喜好程度。[54]也表明潜在变量的引入,如购买的产品意向(例如,给自己的还是送礼的,与工作相关的还是休闲用的等),虽然其真实值是未知的,但可以显式地将其建模在贝叶斯网络(BN)中,的确能够提高BN分类器的预测性能。因此,即使没有任何显式的情境信息(例如,家庭中的哪个成员在收看节目),推荐的准确性仍然可以通过使用精心选择的能够处理潜在变量的机器学习技术来改进(例如,通过在设计良好的推荐模型中使用潜在变量)。文献[11]提出了类似的使用潜在变量方法。

正如7.2.1节中,我们关注Dourish[34]提出的表征性观点,假设情境被定义为一组



预先定义的情境属性,其结构不随时间改变。这个假设的含义是,在做出推荐前,我们需要识别和获取情境信息。如果情境信息获取的过程是显式的,或者甚至是隐式的,也应该作为整体数据收集过程的一部分去执行。所有这些意味着情境信息是相关的,且在设计阶段就要明确需要收集哪些信息,并在未来的实际推荐中良好应用。

Adomavicius 等[3]给出了一种在具体的推荐业务中,对情境属性的选择方法。特别地,[3]给出了大量应该由业务专家初步筛选出,以备后续进一步选择的情境属性。例如,在例1中描述一个电影推荐业务场景,我们可以初步考虑的情境属性,如时间、剧院、伙伴、天气以及其他会影响看电影时的情境属性,这些早期都是由该领域的专家挑选的。然后,收集完评分数据和情境信息的数据后,我们可能会用各种统计学方法测试所选择的情境属性与看电影体验的关联,通常不同的情境属性的评分会表现出显著差异。例如,我们可以应用配对  $t$  检验,看是否好天气与坏天气或一个人看电影和跟同伴一起看是否显著影响看电影的体验(通过评分分布的统计学显著性反映)。这个过程表明,可以通过扫描所有可考虑的情境属性,过滤掉那些在某个具体推荐业务中影响不大的属性。例如,我们可以认为时间、剧院和同伴这些情境属性是有意义的,而天气情况在电影推荐业务场景下则不需要考虑。

### 7.3 结合情境的推荐系统形式

情境信息推荐系统的使用可以追溯到 Herlocker 和 Konstan 的研究成果[36],他们假设在某些领域里,把与用户任务有关的知识纳入推荐算法中能产生更好的推荐。例如,如果我们想推荐那些作为礼物送给孩子的书,那么我们可能要指明哪几本书是小朋友已经拥有的(喜欢的),然后把该信息(即该任务的档案)提供给推荐系统计算新的推荐。注意,这种方法仍然是在传统二维的  $User \times Item$  空间里操作的,因为推荐任务对某个用户的描述只包含了样本物品清单;换句话说,除了标准的  $User$  和  $Item$  的维度,并没有额外的情境维度。然而,这种方法成功地告诉我们如何将额外的相关信息(通过指定与用户和任务相关的物品这种形式)纳入标准的协同过滤体系。此外,根据不同主题打上兴趣分数也被推荐系统用于构建用户情境属性[73]。

在推荐系统中利用情境信息有多种方法,大致可以分为两类:1)通过情境驱动的查询和搜索产生的推荐;2)通过情境偏好提取和估计的推荐。情境驱动的查询和搜索方法已被各种移动和旅游推荐系统采用[2, 27, 68]。使用这种方法的系统通常使用的情境信息(可能是直接询问并获得用户当前的情绪或兴趣,或从环境中获得当地的时间、天气或地理位置)去查询或搜索某些资源库(如餐厅),同时给用户匹配度最好的资源(如附近哪些餐馆在营业)。早期运用这种方法的是 Cyberguide 项目[2],对不同的手机平台开发了几种导游原型。Abowd 等[2]讨论了为给手机用户提供导游服务所需提取的特征,比如,用户当前和过去的地理位置等情境信息就可以在推荐和指引过程产生作用。情境感知导游系统的研究文献还有 GUIDE[31]、INTRIGUE[14]、COMPASS[68]、MyMap[32]等系统。

在推荐流程中还可以通过情境偏好提取和估计进行推荐,这也是当前情境感知推荐系统研究的一种研究趋势,在文献[3, 53, 55, 72]有介绍。与前面所讨论的情境驱动的查询和搜索(推荐系统利用当前情境信息,以及当前用户给定的兴趣作为查询条件去搜索为最合适的内容)不同,使用这种方法的技术试图通过学习来对用户的偏好建模。这可以通过观察系统中用户和其他用户的交互行为,或通过获取用户对以前推荐的物品的偏好反馈。为了对用户的情境敏感属性建模并产生推荐,这些技术会采用基于内容的协同过滤,

或者综合各种情境感知的推荐方法,或者引入数据挖掘和机器学习领域的各种智能数据分析技术(如贝叶斯分类器和支持向量机)。

两种方法都有不少研究上的挑战性,但在接下来本章的重点会是第二种,即近期流行的应用情境偏好提取和估计的研究方法。我们想说明的是,结合两种方法中的技术来设计单一的业务系统是可行的(即无论内容驱动的查询和搜索,以及情境偏好提取和估计)。例如,UbiquiTO系统[27]实现了一个移动导游服务,不仅可以智能地适应特定的情境信息,也可以使用各种规则和模糊集技术调整应用内容以适应用户的偏好和兴趣。同样,News@hand系统[26]使用语义技术提供的个性化新闻推荐,这些推荐是基于对用户查询语句的分析或者根据特定用户(用户组)的属性计算得到的。

接下来讨论情境偏好暗示和估算技术。一般来说,传统的二维( $User \times Item$ )推荐系统可以描述为一个函数,它把用户部分偏好数据作为其输入,产生的每个用户的推荐列表作为输出。这样就得到了图7.3中的传统二维推荐过程的示意图,它包括三个组成部分:数据(输入)、二维推荐器(函数)、推荐列表(输出)。需要注意的是,如图7.3所示,当基于已有数据的推荐函数被定义后(或构造后),将构造好的推荐函数应用在给定用户 $u$ 以及所有候选物品上,以得到每个物品的预测评分,再根据预测评分进行排序,就能获得被推荐的物品。在本节的后面部分,我们将讨论在这三个组成部分中如何使用情境信息来产生三种不同的情境感知推荐系统。

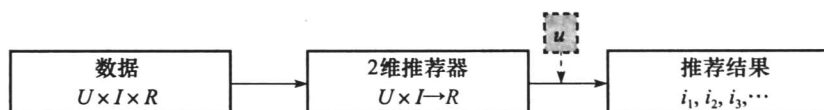


图 7.3 传统推荐流程的一般组成部分

在7.2.2节中提到,传统的推荐系统建立在用户的部分喜好上,即部分物品(限定10个)的用户偏好。传统的推荐系统的输入数据,通常有如 $\langle \text{用户}, \text{物品}, \text{评分} \rangle$ 的格式。相比之下,情境感知推荐系统建立在用户的部分情境偏好知识的基础上,数据格式通常是 $\langle \text{用户}, \text{物品}, \text{情境}, \text{评分} \rangle$ ,每条记录中不仅包括给定用户对特定物品的喜好程度,同时也包括这个用户使用该物品的情境信息(例如,情境=星期六)。此外,除了用户的描述信息(如人口统计学信息),物品的描述信息(如物品的特征)和评分信息(如多标准评分信息),情境感知推荐系统还可以利用额外的7.2.2节中提到的情境属性,如情境层次结构(如星期六 $\rightarrow$ 周末)。由于这些额外的情境数据的存在,带来了几个重要的问题:在对用户偏好建模时如何体现情境信息?我们可否重用在传统(非情境)推荐系统中的知识积累去生成情境感知的推荐吗?我们将在本节详细探讨这些问题。

在提供的已有的情境信息后,如图7.4所示,我们先从具备形式 $U \times I \times C \times R$ 的数据开始,其中 $C$ 是额外的情境维度,并最终为每个用户生成情境推荐列表 $i_1, i_2, i_3, \dots$ 。然而,与未考虑情境信息的图7.3不同,我们可以把当前(或期望)的情境 $c$ 应用在推荐过程的不同阶段。更具体地说,取决于在哪个部分应用了情境信息,基于情境偏好提取和估计的情境感知的推荐流程可以有三种形式,如图7.4所示:

- **情境预过滤(或基于情境的推荐输入)**。在该推荐方式中(图7.4a),情境信息决定了数据选择或数据构建。换句话说,当前的情境信息 $c$ 被用于选择或构建相关的数据记录集(评分)。在完成数据选择后,评分预测可以在任何传统的二维推荐系统完成。
- **情境后过滤(或基于情境的推荐输出)**。在该推荐方式中(图7.4b),情境信息一开

始被忽略,全部数据都使用传统的二维推荐系统预测评分。然后,推荐的结果会针对每个用户的情境信息进行修正。

- 情境建模(或基于情境的推荐函数)。在该推荐方式中(图 7.4c),情境信息直接用于建模,直接影响评分。

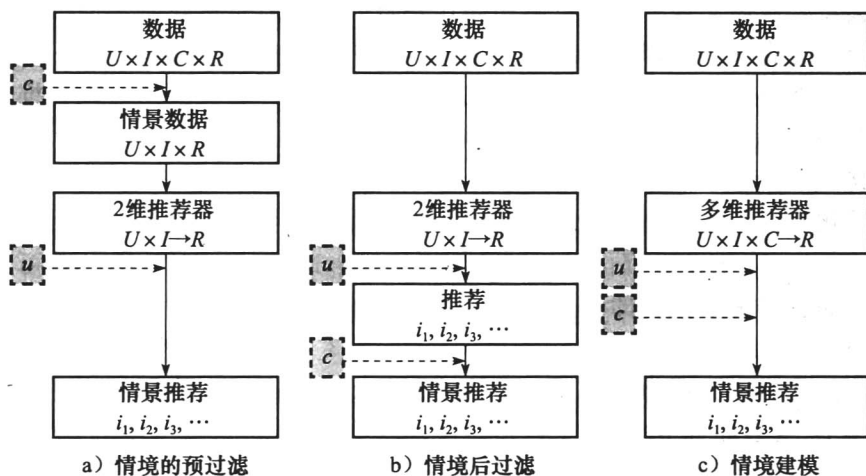


图 7.4 推荐系统中整合情境的方式

在本章其余部分会详细讨论这三种方式。

### 7.3.1 情境预过滤

如图 7.4a 所示,情境预过滤的方法是使用情境信息来选择或构建相关性最强的二维 (User×Item) 数据以用于推荐。这种方法的一个主要优点是,它允许应用任何先前文献[5]中提出的各种传统推荐技术。值得一提的是,这种方法的一个可能用途是情境  $c$  本身可以作为一个查询去挑选(或过滤)有关的评分数据。电影推荐系统的情境数据过滤的一个例子是:如果一个人要周六看电影,推荐电影时仅使用周六电影的评分数据即可。注意,这个例子展示的是一个精确的预过滤器。换句话说,数据筛选查询由具体的情境信息构成。

例如,根据情境预过滤的范式,Adomavicius 等提出了一种基于降维的方法[3],把情境推荐问题从多维(MD)下降为标准的二维 User×Item 空间。因此,对任何情境预过滤方法来说,降维的一个重要好处是,完成降维后,以前所有的二维推荐系统的研究方法可以直接应用在高维场景中。具体来说,令  $R_{User \times Item}^D: U \times I \rightarrow Rating$  为任意二维评分估算函数,给定现有的评分  $D$ (即  $D$  包含所有已知的、用户定义的评分记录  $\langle user, item, rating \rangle$ ),我们也能预测任意物品的评分,如  $R_{User \times Item}^D(John, StarWars)$ 。类似地,一个支持时间情境的三维评分预测函数类似的定义如下:

$$R_{User \times Item \times Time}^D: U \times I \times T \rightarrow Rating$$

其中,  $D$  包含用户定义的评分记录  $\langle user, item, time, rating \rangle$ 。这样就可以用一些方法将三维预测函数表达成二维预测函数了,例如:

$$\forall (u, i, t) \in U \times I \times T, R_{User \times Item \times Time}^D(u, i, t) = R_{User \times Item}^{D[Time=t]}(u, i)$$

这里  $D[Time=t]$  表示一个简单的情境预过滤器,而  $D[Time=t](User, Item, Rating)$  表示从  $D$  里面选择 Time 维度值为  $t$  的数据,且只保留 User 和 Item 两个维度,以及评分本身的值。也就是说,如果我们把一个三维评分数据集  $D$  当成一种关系,那么  $D[Time=$

$t](User, Item, Rating)$ 是从  $D$  执行了选择和投影两个关系运算后得到另一个关系。

但是,确切的情境有时会过于狭隘。举个例子,跟女朋友在周六去影院观看电影的情境,即  $c=(\text{女友}, \text{剧院}, \text{周六})$ 。使用如此确切的情境作为数据过滤条件可能是有问题的,这里有几个原因。首先,某些方面过于具体的情境可能并不是很重要。例如,用户带着女朋友在周六去影院看电影的偏好,可能和周日完全一样,但是跟周三不同。因此,宽泛一点的情境可能更适合一些,即周末,而不是周六。此外,在确切的情境下可以被用来做评分预测的数据可能很少,这在推荐系统领域称为“稀疏性”问题。换句话说,推荐系统中可能并没有很多用户带上女朋友周六去看电影的历史数据。

**情境泛化。**Adomavicius 等介绍了泛化预过滤的概念[3],允许把具体的情境数据在预过滤查询中泛化。定义: $c'=(c'_1, \dots, c'_k)$ 是情境信息  $c=(c_1, \dots, c_k)$ 的泛化,当且仅当  $c_i \rightarrow c'_i$  对于每个  $i=1, \dots, k$  都位于相同的情境层级。这样, $c'$ (而不是  $c$ )就可以被用作数据检索条件从而获得基于情境的评分数据。

根据情境信息泛化的概念,Adomavicius 等[3]提出不要使用简单的预过滤器  $[Time=t]$ ,这里的  $t$  是评分  $(u, i, t)$  的确切情境,而应该使用一般化的预过滤器  $[Time \in S_t]$ ,  $S_t$  代表情境信息  $t$  的一个集合。 $S_t$  在这里被称作情境分块。例如,如果我们想预测 John Doe 在周一有多想看“角斗士”,即计算  $R_{User \times Item \times Time}^D(\text{JohnDoe}, \text{Gladiator}, \text{Monday})$ ,我们可以不用用户给出的 Monday 评分去做预测,而是可以使用 Weekday 这种泛化的评分。换句话说,对于每个  $(u, i, t)$  中的  $t \in \text{Weekday}$ ,我们可以通过  $R_{User \times Item \times Time}^D(u, i, t) = R_{User \times Item}^{D[Time \in \text{Weekday}]}(User, Item, AGGR(Rating))(u, i)$  预测评分。更一般地,为了预测某些评分  $R(u, i, t)$ ,我们可以使用某些特定的情境信息分块  $S_t$ :  $R_{User \times Item \times Time}^D(u, i, t) = R_{User \times Item}^{D[Time \in S_t]}(User, Item, AGGR(Rating))(u, i)$ 。

注意,我们已经在上面的表达式使用了  $AGGR(Rating)$ (即聚合函数)的概念。这是因为,在数据集  $D$  的同一情境分块  $S_t$  中,在不同的时间下同一用户对同一物品可能会有不同的评分(例如,同样的用户,同样的电影在 Monday 和 Tuesday 得到的评分不同,但都属于工作日分段)。因此,我们必须使用聚合函数来聚合值,例如,在对推荐空间降维时计算评分的均值。上述三维降维方法可以推广到一般的预过滤方法,将任意  $n$  维推荐空间降维到  $m$  维(其中,  $m < n$ )。在本章中,我们假定  $m=2$ ,因为传统的推荐算法是针对二维的推荐空间  $User \times Item$  设计的。注意,由于情境分类的不同和所需要的情境粒度的不同,通常存在多种情境泛化的方案。例如,假设我们有以下的情境分类(is-a 或 belongs-to 关系)层次:

- 同伴: 女朋友  $\rightarrow$  朋友  $\rightarrow$  熟人  $\rightarrow$  任意伙伴;
- 地点: 剧院  $\rightarrow$  任意地点;
- 时间: 星期六  $\rightarrow$  周末  $\rightarrow$  任意时间。

那么下面给出一些针对情境信息  $c=(\text{女朋友}, \text{影院}, \text{星期六})$  可能被泛化的一些例子:

- $c'=(\text{女朋友}, \text{任意地点}, \text{星期六})$ ;
- $c'=(\text{朋友}, \text{影院}, \text{任意时间})$ ;
- $c'=(\text{熟人}, \text{影院}, \text{周末})$ ;

因此,选择“正确”的泛化预过滤器成为一个重要的问题。一种方法是基于经验人工选择,例如,每周的某一天可以泛化为平日或周末。另一种方法更自动化一些,对每个泛化过滤器输出结果的预测效果在過去的数据上进行评估,然后自动选择一个评估中预测效果最好的泛化过滤器。如何处理由于情境粒度导致的计算复杂度是一个有意思的研究课题。

换句话说,在情境粒度很细的业务场景里,会存在大量可能的情境泛化。这时基于穷举的技术是不太可行的,需要更有效的贪婪算法。相关的工作有 Jiang 和 Tuzhilin[39]研究的最优顾客分群粒度使得预测效果最优的问题。在情境感知推荐系统场景下应用这些技术是今后研究的一个方向。

要指出的是,这种降维方法还跟机器学习和数据挖掘中建立局部模型的问题有关[10]。降维方法建立局部的评分估算模型,只使用用户指定的条件下(如早晨)的评分进行推荐,而并非建立全局性评分估算模型,利用所有已有的评分进行推荐。采用降维方法很关键的一点是,要弄清楚在忽略所有与情境无关的信息之后,局部模型的推荐效果是否真的优于传统的二维全局模型。例如,使用情境预过滤推荐周末去电影院看电影可能比较合适,但在家用录像机看电影的推荐还是使用传统的二维技术比较好。这是因为降维方法仅关注了特定的一部分数据,也只利用这部分数据建模并进行推荐,从另一个角度看,只利用部分数据建模会限制推荐物品的广度。所以在这里需要进行权衡:在计算未知评分时,是参考较多的同类情境数据,还是只参考某个特定局部的数据(后者会导致稀疏性问题)。这也解释了降维方法在某些时候比传统的二维推荐技术好,然而在另一些时候却表现不佳的现象。在实际应用中如何选择这两个方案取决于业务领域和可用的数据。基于这一观察,Adomavicius 等[3]提出了将情境预过滤器与传统的二维技术相结合的方法(即将传统的二维技术视为通过了一个默认的、并不执行过滤的过滤器),这种做法将作为案例研究在 7.4 节进行讨论。

最近的一些研究中,Ahn 等[8]使用类似情境预过滤的技术向移动用户推送广告,其参考的因素是用户的位置、兴趣、时间;Lombardi 等[49]通过分析采用了预过滤方法进行推荐的在线零售商的业务数据来评估情境推荐的效果。此外,Baltrunas 和 Ricci[16]提出了一个和情境预过滤稍有不同的方法,该方法引入了一种被称为“物品切分”(item splitting)的技术,这种技术可根据情境的不同将物品切分为若干“虚拟物品”(fictitious item),以找出最适合当前情境的虚拟物品并进行推荐。类似物品切分的思路,Baltrunas 和 Amatriain[15]引入微属性(或用户属性切分)概念,即将用户属性切分成几个子属性(可能有重叠),每个微属性都代表了特定用户在特定情境的特点。预测则通过这些情境下的微属性实现,而不是一个单一的用户模型。这些数据构建技术非常适合情境预过滤方式,因为他们遵循相同的基本思路(如前所述)——利用情境信息把多维推荐问题降维到标准的二维 User×Item 空间,从而允许使用任何传统的推荐评分预测技术。

### 7.3.2 情境后过滤

如图 7.4b 所示,情境后过滤方法在进行推荐时忽略输入数据的情境信息,即在对候选物品进行评分并排序后,再从中任意选择前  $N$  个进行推荐,这里  $N$  是预先设定的。之后,情境后过滤方法使用情境信息为每个用户调整获得的推荐列表。推荐列表的调整有以下方法:

- (在给定的情境下)过滤无关的推荐;
- (基于给定的情境)调整列表里推荐的排名。

例如,在电影推荐业务中,如果一个人想在周末看电影,而且在周末她只看喜剧,那么系统可以过滤掉推荐列表中所有非喜剧的电影。更一般地,情境后过滤方法的基本思想是在给定场景下分析给定用户的情境偏好数据,发现特定物品的使用模式(例如,用户 Jane Doe 在周末只看喜剧),然后用这些模式来调整物品清单,从而产生更“情境相关”的推荐。如图 7.5 所示。



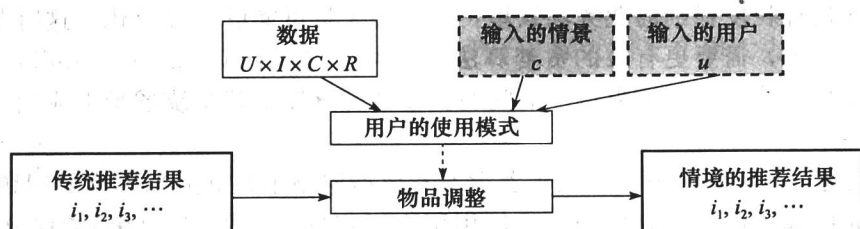


图 7.5 情境后过滤方法的最后阶段：推荐列表的调整

如同其他的推荐技术，情境后过滤方法可分为启发式和基于模型的技术。启发式后过滤方法的重点是对给定的用户在特定情境（例如，某些情况下会选择喜欢的演员）发现物品的共同特征（属性），然后使用这些属性来调整推荐。这其中包括：

- 过滤那些推荐物品中关键特征不足的推荐（例如，被推荐的电影须在给定的情境下有至少两个用户喜欢的演员）。
- 利用被推荐的物品中包含相关的属性的数目给这些物品进行排名（例如，在特定场景下，一部电影包含越多用户喜爱的明星排名越高）。

另一方面，基于模型的后过滤方法可以通过建立预测模型来计算用户在给定情境下选择特定类型物品的概率（例如，在给定情境下选择某种类型电影的可能性），然后用这个概率调整推荐，包括：

- 过滤那些相关性概率小于预定的最小阈值的推荐物品（例如，删除那些被选中机会较小的电影类型）。
- 对预测评分和相关性概率加权之后，给出被推荐物品的排名。

Panniello 等[55]在真实的电子商务数据集上对预过滤方法（7.3.1 节中讨论）和两个不同的后过滤方法——加权和过滤——进行了实验比较。加权后过滤方法将给定情境的相关性概率作为权重计入推荐评分中，并据此对推荐物品重新排序，而过滤后过滤方法直接过滤掉与给定情境相关性概率较小的推荐结果。实验结果表明，加权后过滤方法优于预过滤方法，而预过滤方法优于过滤后过滤方法，因此，方法（预过滤或后过滤）的好坏取决于给定的业务。

与情境预过滤方法类似，情境后过滤方法的一个主要优点是它允许使用任何已有的传统推荐技术[5]。此外，与情境预过滤方法一样，如何将情境泛化技术结合到后过滤方法中是一个值得研究的问题。

### 7.3.3 情境建模

如图 7.4c 所示，情境建模方法直接在推荐函数中把情境信息作为预测用户对物品评分的显式因素来考虑。与情境预过滤和后过滤都可以使用传统的二维推荐函数不同，情境建模的方法生成的是真正的多维推荐函数。它可以直接生成一个预测模型（通过决策树、回归、概率模型等技术），或将情境信息与用户物品数据整合在一起进行启发式计算（即  $\text{Rating} = R(\text{User}, \text{Item}, \text{Context})$ ）。在过去的 10~15 年中，人们基于启发式或其他预测建模技术研究了大量的推荐算法，其中的一些可以从二维拓展到多维推荐环境。我们将在接下来的几个小节介绍几种可被用于情境建模的方法[4]。

#### 7.3.3.1 启发式的方法

传统的基于邻域的二维方法[21, 62]可以通过一种简单的技术直接推广到包含情境信息的多维情形：直接计算  $n$  维距离，而非传统的计算用户—用户或物品—物品相似度的技术。

我们以 User×Item×Time 推荐空间为例,来看看这是怎么做到的。在这里,基于相关评分加权求和的传统最近邻启发式预测函数  $r_{u,i,t}$  可表示为(其他细节可参考文献[4])

$$r_{u,i,t} = k \sum_{(u',i',t') \neq (u,i,t)} W((u,i,t), (u',i',t')) \times r_{u',i',t'}$$

其中,  $W((u,i,t), (u',i',t'))$  描述评分  $r_{u',i',t'}$  对于预测  $r_{u,i,t}$  的权重,  $k$  是归一化因子。权重  $W((u,i,t), (u',i',t'))$  通常都是跟点  $(u,i,t)$  和点  $(u',i',t')$  的在多维空间的距离,即  $\text{dist}[(u,i,t), (u',i',t')]$ , 成反比。换句话说,两个点越近,  $r_{u',i',t'}$  的权重越高。 $W$  可以取为  $W((u,i,t), (u',i',t')) = 1/\text{dist}[(u,i,t), (u',i',t')]$ , 也可以有其他各种表达形式。通常来说,距离度量  $\text{dist}$  的选择很可能基于特定的情境。其中一种简单方式是用降维方法定义一个多维  $\text{dist}$  函数(有点类似于 7.3.1 节的描述): 只考虑相同情境信息下的数据, 即

$$\text{dist}[(u,i,t), (u',i',t')] = \begin{cases} \text{dist}[(u,i), (u',i')], & \text{若 } t = t' \\ +\infty, & \text{其他} \end{cases}$$

这个距离函数使得  $r_{u,i,t}$  仅依赖于具有相同时间值  $t$  的数据。所以在这种情况下, 我们只用和被预测评分具有相同情境  $t$  的数据  $(u,i,t)$  评分作为依据, 将评分函数降维为二维函数。此外, 如果我们进一步细化距离函数  $\text{dist}[(u,i), (u',i')]$ , 使其只依赖于当  $i = i'$  时用户之间的距离, 那么我们将获得类似于之前描述的预过滤方法。而且这种方法很容易扩展到任意一个多维的情况, 只需要设定两个情境  $t$  相同的数据之间的距离  $\text{dist}[(u,i), (u',i')]$  即可。

另外, 还有一些其他方法来定义距离函数, 比如, 使用加权曼哈顿距离:

$$\text{dist}[(u,i,t), (u',i',t')] = w_1 d_1(u, u') + w_2 d_2(i, i') + w_3 d_3(t, t')$$

或加权欧几里得距离:

$$\text{dist}[(u,i,t), (u',i',t')] = \sqrt{w_1 d_1^2(u, u') + w_2 d_2^2(i, i') + w_3 d_3^2(t, t')}$$

这里  $d_1$ 、 $d_2$  和  $d_3$  是分别是用户、物品和时间的距离函数,  $w_1$ 、 $w_2$  和  $w_3$  是对每一个维度分配的权重(可以根据它们的重要性来确定权重)。总之, 距离函数  $\text{dist}[(u,i,t), (u',i',t')]$  可以定义成许多不同的形式, 而在许多系统中, 它通常只计算相同用户或相同物品的评分。研究距离函数的各种形式, 并比较这些不同距离定义对预测性能的影响是一个有意思的研究课题。

### 7.3.3.2 基于模型的方法

针对传统的二维推荐模型, 现已有一些相关推荐技术的研究[5]。其中的一些方法可直接扩展到多维的情况[12], 该文的二维技术优于一些先前已知的协同过滤方法。

Ansari 等[12]将用户和物品的信息结合起来, 建立了一个统一的基于回归、且有层次的贝叶斯偏好模型, 并使用马尔可夫链蒙特卡罗(MCMC)技术来估算该模型的参数。具体来说, 我们假设有  $N_U$  个用户, 每个用户  $u$  由该用户已观察属性组成的矢量  $z_u$  定义, 如性别、年龄、收入。还假设有  $N_I$  个物品, 每个物品  $i$  由该物品的属性向量  $w_i$  定义, 如价格、质量和大小。假设  $r_{ui}$  为用户  $u$  对物品  $i$  的评分, 其中  $r_{ui}$  是一个实数。而且评分  $r_{ui}$  只在部分(用户、物品)对上定义。未知评分的估计问题可以定义为

$$r_{ui} = x'_{ui}\mu + z'_u\gamma_i + w'_i\lambda_u + e_{ui}, e_{ui} \sim N(0, \sigma^2), \lambda_u \sim N(0, \Lambda), \gamma_i \sim N(0, \Gamma)$$

模型中已观察的(已知的)值包括: 用户  $u$  对物品  $i$  的评分  $r_{ui}$ , 用户属性  $z_u$ , 物品属性  $w_i$ , 以及向量  $x_{ui} = z_u \otimes w_i$ , 这里  $\otimes$  是克罗内克积, 即一个包含单个元素  $z_u$  和  $w_i$  的所有可能的叉乘组合的长向量。直观地说, 这个公式提出了一个用户  $u$  的特征  $z_u$ , 物品  $i$  的特征

$w_i$ , 以及用户和物品的交互作用(interaction effect)  $x_{ui}$  来描述未知评分  $r_{ui}$  的回归模型。交互作用源自模型的层次结构, 其用以捕捉类似如用户的年龄如何影响他对某些电影流派的偏好之类的特征。上述方程中的向量  $\mu$  表示回归模型中的未观察(未知)的因素(指交互作用)的系数(斜率), 类似下面将会讨论的回归模型中需要估计的其他未知系数。向量  $\gamma_i$  代表物品  $i$  的权重系数, 由物品  $i$  的特质决定, 反映了未在物品  $i$  的档案中记录的其他属性对最终推荐评分的影响, 如导演、音乐和电影中的演技。同样, 矢量  $\lambda_u$  代表特定用户  $u$  的权重系数, 由用户的特质决定, 反映了用户  $u$  未在用户档案中记录的其他属性对最终推荐评分的影响。最后, 误差项  $e_{ui}$  是均值为 0、标准差  $\sigma$  的正态分布。此外, 我们考虑一个层次模型, 并假设回归参数  $\gamma_i$  和  $\lambda_u$  服从均值为 0, 协方差分别为  $\Gamma$  和  $\Lambda$  的正态分布, 记为  $\gamma_i \sim N(0, \Gamma)$ ,  $\lambda_u \sim N(0, \Lambda)$ , 其中  $\Gamma$  和  $\Lambda$  未知。综上所述, 该评分模型的参数是  $\mu$ 、 $\sigma^2$ 、 $\Lambda$  和  $\Gamma$ 。这些参数可以用文献[12]中描述的 MCMC 方法从已知评分数据中推导出来。

尽管文献[12]提出的方法仅针对传统的二维推荐系统, 它也可以直接推广到情境信息推荐系统。例如, 假设我们有第三个维度 Time, 由以下两个属性(变量)定义: 1) 布尔值 weekend 表明某一部片子是不是在周末观看的及; 2) 一个正整数 numdays 表明观看这部片子时离片子上映的天数。

在这种情况下, Ansari[12]模型可以扩展到三维(时间), 相关文献可参考 Adomavicius 和 Tuzhilin[4]:

$$r_{uit} = x'_{uit}\mu + p'_u\theta_i + q'_u\lambda_u + r'_u\gamma_i + z'_u\delta_u + w'_i\pi_u + y'_i\sigma_{ui} + e_{uit}$$

式中,  $e_{uit} \sim N(0, \sigma^2)$ ,  $\gamma_i \sim N(0, \Gamma)$ ,  $\lambda_u \sim N(0, \Lambda)$ ,  $\theta_i \sim N(0, \Theta)$ ,  $\delta_u \sim N(0, \Delta)$ ,  $\pi_u \sim N(0, \Pi)$ ,  $\sigma_{ui} \sim N(0, \Sigma)$ 。

该模型包括了已观察和未观察的用户, 物品和时间以及它们之间的交互对  $r_{uit}$  的影响,  $r_{uit}$  是用户  $u$  在时间  $t$  对电影  $i$  的评分。变量  $z_u$ 、 $w_i$  和  $y_i$  代表用户的观察值(如自然属性), 电影的观察值(如流派)和时间维(如周末、上映天数)。矢量  $x_{uit}$  代表用户、电影和时间变量的交互作用, 其系数  $\mu$  表示未观察到(未知)的回归直线(即交互作用的回归线, 交互作用都是隐式的)的斜率, 即在上面的回归模型中需要估计的未知系数, 类似以下其他参数。向量  $\lambda_u$ 、 $\gamma_i$  和  $\theta_i$  代表未观测到的其他属性——用户(如他们的种族)、电影(如故事、剧本等)和时间(例如, 是否在假期观看电影, 电影上映的季节等)——的影响。向量  $p_u$  表示观察到的用户和物品之间的交互作用, 类似定义  $q_u$  和  $r_u$ 。向量  $\sigma_{ui}$  代表未观察到的用户和物品交互作用, 类似定义向量  $\pi_u$  和  $\delta_u$ 。综上所述, 模型的参数为  $\mu$ 、 $\sigma^2$ 、 $\Lambda$ 、 $\Gamma$ 、 $\Theta$ 、 $\Delta$ 、 $\Pi$  和  $\Sigma$ , 可以使用马尔可夫链蒙特卡罗方法(MCMC)[12]。

最后, 请注意, 在上述模型中的需要估计的参数数量会随着维度的增加而增加, 因此, 评分的稀疏性很有可能是一个问题。如果在某些业务场景下这会是严重的问题, 所以可忽略上述模型中的某些项, 以形成一个简化模型。例如, 我们可能会忽略  $q_u\lambda_u$ , 或者其他一些项, 从而得到一个参数较少的简单模型。需要注意的是, 这个简单的模型仍然会考虑情境信息, 如时间。此外, 该模型的参数估计非常耗时, 且不可扩展。因此, 研究的挑战之一是使此类模型对未知评分的估计更具扩展性且更稳定。如何使这些方法更具扩展性的一些初步想法可参阅文献[67]。

除了将现有的二维推荐技术扩展到多维, 也出现了一些专门为基于情境模型范式的情境感知推荐系统开发的新技术。例如, 遵从一般的情境建模范式, Oku 等[53]直接将额外的情境维度(如时间、伴侣和天气)整合到推荐空间, 并使用机器学习技术为餐厅推荐系统提供推荐。特别的是, 他们使用支持向量机(SVM)分类方法, 把在不同情境下, 顾客喜

欢的物品和不喜欢的物品，看作多维空间的两个向量集，并在这个空间中构造一个分割超平面，使得两个数据集之间有最大的距离。所产生的超平面就是下一步推荐决策的分类器（如果物品落在超平面上的“喜欢”的一面，会给出推荐，如果它落在了“不喜欢”的一面，则不会推荐）。此外，Oku 等[53]经验显示，情境感知 SVM 在推荐准确度和用户满意度上明显优于非情境的基于 SVM 的推荐算法。类似地，Yu 等[72]使用情境建模方法，通过引入情境中作为额外的模型维度，混合利用各种推荐技术（综合了基于内容的贝叶斯分类器和基于规则的方法）为智能手机用户提供内容的推荐。

最后，文献[1]给出的个性化的访问模型(PAM)同样也是基于模型的，它提供了一套个性化的服务，包括情境信息发现、语境化(contextualization)、结合(binding)和匹配(matching)服务。Abbar 等[1]描述了如何组合这些服务以形成情境感知推荐系统(CARS)，并给出了实现，以提供优质的情境感知的推荐。

本节在预过滤、后过滤、情境建模方法的框架内描述了各种将情境信息用到推荐算法中的方法。由于 CARS 是推荐系统中的一个新兴领域，本节介绍的这些方法只是能给出推荐的初步方法，更好的方法需要且应该进一步发展这三种基本方法。

在 7.4 节将讨论如何将这些不同的方法结合在一起。

7.4 多种方法结合

许多推荐系统文献都表明，综合几种解决方案（“混合”使用或“组装”使用）通常比单一方法更能提高性能[24, 25, 43, 56]。情境感知推荐系统的三个范式为合并入新的方法提供了几种不同的方式。

一种可能的方式是结合相同类型的几种模式。例如，Adomavicius 等[3]按照这种方法将几个不同的情境预过滤器结合在了一起。采用多个预过滤器的原因基于如下事实：如前所述，对相同特定情境通常可以有多个不同的（和可能相关的）泛化。例如，情境  $c = (\text{女友}, \text{剧院}, \text{周六})$  可以泛化为  $c_1 = (\text{朋友}, \text{任意地点}, \text{星期六})$ ， $c_2 = (\text{熟人}, \text{剧院}, \text{任意时间})$ ，以及一些其他的情境。根据这个想法，Adomavicius 等[3]根据每个评分可能的情境的数目设定预过滤器的数目，再将各个预过滤器的推荐结果组合起来。这种方法的示意图参见图 7.6。需要注意的是，组合多个预过滤器，可以有多种方法。例如，对于一个给定的情境，1) 可以选择表现最好的预过滤器；2) 使用预过滤器的“组合”。在 7.4 节的其余部分将研究 Adomavicius 等的案例[3]。

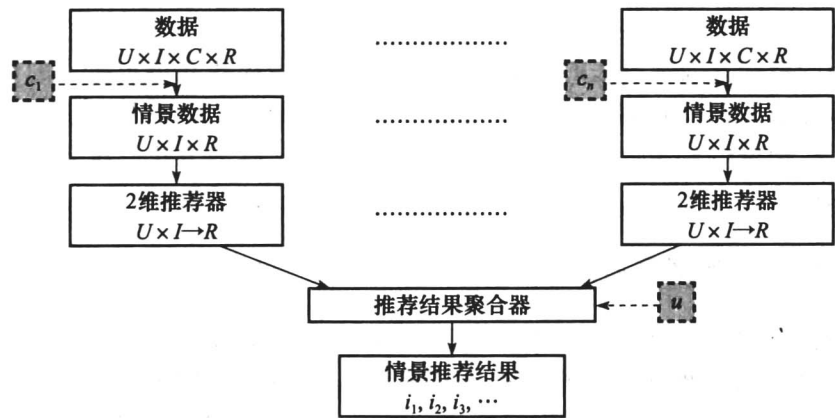


图 7.6 多预过滤器的组合：整体示意

另一个有趣的地方可能来自于复杂的情境信息通常可被分成几个组成部分,每个情境信息组成部分的作用可能是不同的,从而有不同的实现选择:是用在预过滤、后过滤,还是建模阶段。例如,时间信息(平日与周末)可能在预过滤阶段最有用,但天气信息(晴天与阴雨)可能在后过滤阶段最有用。确定不同的情境信息在情境感知推荐系统的最适用模式,也是一个有趣的和有前途的研究方向。

#### 7.4.1 组合预过滤器案例研究:算法

组合评分估算包括两个阶段[3]:1)使用已知的用户指定的评分(即训练数据),选择优于传统协同过滤方法的预过滤器;2)为了在一个给定的情境下预测评分,选择该情境下最佳的预过滤器,并在这个片段上使用二维的推荐算法。

第一阶段是预处理阶段,通常“离线”执行,它可以与任何传统的二维评分估计方法  $A$  一起使用,包括以下三个步骤[3]:

- 在所有可能的泛化预过滤器中,找到那些包含大量数据的情境分块,即分块有超过  $N$  个以上的评分,其中  $N$  是某个预定的阈值(例如,在该案例中  $N=250$ )。如果推荐的空间比较“小”(指维度的数量和每个维度的属性范围),可以简单地对空间的所有可能的分块进行穷举搜索(指遍历尝试各种泛化规则)以获得大的分块。另外,领域专家(如市场部经理)或一些贪婪的启发式算法都有助于寻找业务中重要的大分块。
- 算法  $A$  使用每个在上一步中确定的泛化预过滤器  $c'$  进行预测,其预测性能由选定的性能指标确定(本案例使用  $F$  值(F1-measure),被定义为准确率和召回率的调和平均数。后者是两个标准的,广泛应用于信息检索的决策支持度量,最近也被引入推荐系统[37])。只有那些在相同的数据分块下,算法  $A$  进行情景预过滤后性能超过算法  $A$  的标准版,即未进行预过滤的,这样的预处理器才会保留下来。
- 对其他预过滤器,如果存在预过滤器  $c'$  和  $c''$  使得  $c' \rightarrow c''$  (即  $c''$  的泛化程度比  $c'$  要高),且算法  $A$  使用预过滤器  $c''$  的性能比使用预过滤器  $c'$  更好,那么  $c'$  被视为多余的(泛化程度不够,且不够精确),并在预过滤器集合中将其去掉。剩下的情境分块集合,记作  $SEGM^*$ ,是上述组合方法“离线”阶段的工作结果。

一旦得到了一套高性能的预过滤器  $SEGM^*$ ,我们可以进入组合评分估算的第二阶段:当需要进行推荐时,从中选择一个合适的预过滤器进行“实时”推荐。给定推荐的具体情境  $c$ ,算法  $A$  采用的最好的预过滤器  $c' \in SEGM^*$  满足  $c \rightarrow c'$  或  $c = c'$ 。如果不存在这样的预过滤器  $c'$ ,那么标准的二维非过滤算法  $A$  (即在整个数据集上训练)也可用于预测评分。

本章介绍的组合预处理方法的主要优点是,它只使用那些在特定情境的情况下优于标准的二维推荐算法的预处理器,假如预处理器没有效果,就继续使用标准的二维推荐算法。因此,可以期望组合预处理方法在实践中会优于纯二维的方法。组合预处理方法要超越二维方法取决于许多不同的因素,如问题的领域或数据的质量。

#### 7.4.2 组合预过滤器案例研究:实验结果

文献[3]给出了在一个真实的电影推荐系统上分别采用组合预过滤器的与传统的二维协同过滤(CF)方法进行性能对比的例子,来评估 7.4.1 节介绍的组合方法的表现。在此业务中,除了要求用户评价自己的电影观赏体验,用户还会要求说明:1)时间(time):什么时候看了这部片子(选择:平日、周末、不记得),此外,如果在周末看的,是否是电影的



首映周末(选择：是、否、不记得)；2)地点(place)：看这部片子的地点(选择：在电影院、在家里、不记得)；3)同伴(companion)：和谁看这部片子(选择：自己一个人、和朋友、和男朋友/女朋友、和家人或其他人)。在过去的 12 个月(2001 年 5 月到 2002 年 4 月)共收集了 117 名学生的 1755 个评分。由于一些学生只对很少的电影给出了评分，忽略掉给少于 10 个电影进行评分的学生的数据。因此，从最初的数据获得最终的数据集只有 62 个学生、202 部电影和 1457 个评分。

在预过滤器选择阶段中，9 个大的情境分块被提取出来，如表 7.1 所示。然而，在这些分块上用标准的协同过滤技术和情境预过滤协同过滤技术进行性能比较，发现只有 4 个“高表现”的分块；其中一个没能通过冗余校验(即 7.4.1 节中所描述的预过滤器的选择阶段的步骤 3)。其余三个高表现的预过滤器如表 7.2 所示，即  $SEGM^* = \{Theater-Weekend, Theater, Weekend\}$ 。

表 7.1 预过滤器的选择阶段步骤 1 产生的大的情境分块

名称	大小	描述
Home	727	Movies watched at home
Friends	565	Movies watched with friends
NonRelease	551	Movies watched on other than the opening weekend
Weekend	538	Movies watched on weekends
Theater	526	Movies watched in the movie theater
Weekday	340	Movies watched on weekdays
GBFriend	319	Movies watched with girlfriend/boyfriend
Theater-Weekend	301	Movies watched in the movie theater on weekends
Theater-Friends	274	Movies watched in the movie theater with friends

表 7.2 高表现值的大情境分组

分段	CF: 分段训练的 F 值	CF: 全体数据训练的 F 值
Theater-Weekend	0.641	0.528
Theater	0.608	0.479
Weekend	0.542	0.484

最后，高表现的分组  $SEGM^*$  应用于“在线”评分估算。总体而言，二维协同过滤和组合预过滤协同过滤方法(受限于相关数据的稀疏性，组合过滤方法并不能对所有场景进行预测)都能对其中的 1373 个场景进行评分预测(共收集了 1457 个)。结果表明，合并后的预过滤协同过滤方法大幅优于传统的二维协同过滤(表 7.3 第一行)。

表 7.3 基于 F 值的整体表现

对比	全体 F-值		F 值分差
	标准 2 维 CF	组合降维 CF	
所有的预测评分	0.463	0.526	0.063
来自 $SEGM^*$ 的评分	0.450	0.545	0.095

需要注意的是，如前所述，组合预过滤协同过滤方法会使用标准的二维协同过滤来预测任何不属于高表现预过滤器的评分。因此在本案例中，这两种方法对不属于分块(Theater-Weekend, Theater, Weekend)的预测结果是相同的。由于这种评分没有体现两种方法

的差别,所以判断两种方法在 SEGM\* 上的评分差异就很重要了。在这种情况下,SEGM\* 包含 743 个评分(共有 1373 个),这两种方法的  $F$  值差异是 0.095(表 7.3 中的第 2 行),比前面描述的场景提升幅度更大。

在本节中,我们讨论了组合多个预过滤、后过滤和情境建模方法去产生更好的预测,主要侧重于组合多个预过滤器,基于文献[3]。我们只列出一些主要的想法,而把大部分的问题作为研究课题留待研究。我们相信通过创造地组合来自机器学习的各种技术,可以显著提高 CARS 的性能,这也是一个重要而有趣的研究领域。

## 7.5 情境感知推荐系统的其他问题

除了将情境整合到推荐系统的三个范式以及将这三个范式相结合的方法,情境感知推荐系统还有其他几个重要的课题,比如,如何更好地利用情境信息,如何让 CARS 具备更丰富的交互能力从而做出更灵活的推荐,以及如何打造高性能 CARS 系统。在本节的其余部分,我们将讨论这些问题。

在预过滤、后过滤以及情境建模方法中进行权衡,对组合它们给出更好的解释。在 7.3 节中,我们只介绍三种一般的 CARS 范式,并没有考虑它们之间的平衡。为了更好地理解预过滤、后过滤和情境建模方法,研究比较了这三种方法,重点了解了这些方法的相对优点和缺点。文献[55]给出了这样的研究,对预过滤方法与一类后过滤方法在推荐的质量上进行比较。结果表明,在  $F$  值意义下,两者在提供更好的推荐方面不分高下。此外, Panniello 等[55]给出了一种条件判定方法,该条件满足时,应使用预过滤而不是后过滤方法。

文献[55]的工作只是一个比较预过滤、后过滤和情境建模方法的系统方案的第一步,为了了解三种方法的优缺点,以及在何种条件下,执行哪种方法更优,还有很多工作需要做。

同样,需要做更多的工作去更好地理解 7.4 节讨论的组合方法。这是一个富有成果的研究领域,对各种预过滤、后过滤和情境建模方法的组合方式还有很大的改进和提升空间。

让 CARS 的交互能力更丰富,推荐更具弹性。情境感知推荐有以下两个重要的性质:

- **复杂性。**由于 CARS 在推荐过程中不仅涉及用户和物品,还涉及各种不同类型的情境信息,这种推荐比传统的不包含情境的推荐,可能会复杂很多。例如,在电影推荐业务中,一个用户(如汤姆)可能会需要推荐最适合他和他女友在周末合适的时间观看的前 3 部电影。
- **交互性。**情境信息通常需要用户在 CARS 中给出。例如,要利用现有的情境信息,CARS 系统在提供针对情境的推荐前,可能需要用户(汤姆)给出他希望和谁看电影(如女朋友),什么时间(如在周末)等信息。

这两个特性的结合会需要更灵活的推荐方法出现,让用户可以表达感兴趣的推荐类型,而不是生搬硬套到大多数当前厂商所提供的只将 top- $N$  物品推荐给用户,反之亦然。交互性的第二个要求,还要求开发出支持用户以交互和迭代的方式在推荐过程中输入信息,且定义良好的用户界面(UI)。

这种灵活的情境感知的建议,可以从几个方面来实现。首先, Adomavicius 等[6]开发了推荐查询语言 REQUEST<sup>⊖</sup>,它允许用户以灵活的方式在相当广泛的推荐范围内刻画自己的个性化需求,从而更准确地反映自身的兴趣。REQUEST 基于在 7.2.2 节和文献[3]

⊖ REQUEST 是推荐查询说明(Recommendation QUery Statement)的缩写。

中描述的多维情境推荐模型,支持多种特性,有兴趣的读者可以在文献[6]中找到这些特性的详细说明、正式的语法以及这种语言的各种性质。此外,Adomavicius等[6]讨论了REQUEST的表达能力给出了一种多维推荐代数学,为该语言提供了理论依据。

到目前为止,我们仅简要提及的建议查询语言本身。由于引进这样一种语言的一个主要论据是其最终用户的使用,所以开发出简单、友好、简洁且能支持灵活但往往复杂的情境推荐的用户界面(UI)也是很重要的。高品质的用户界面应简化用户与推荐系统之间交互的复杂性,使其能用于更广泛的用户。开发这样的用户界面是未来研究的一个课题。

提供灵活推荐的另一个方案,可以看看文献[44]描述的FlexRecs系统及其框架。FlexRecs方法通过把推荐过程从执行过程解耦出来,以实现基于结构化数据的灵活推荐。特别地,推荐可以抽象地表达为一个高层次的含参数的工作流,它包含了传统的关系运算符和新的推荐相关的操作符。

除了用开发语言来表达情境感知推荐,提供能够以交互的方式表达灵活推荐建议的用户界面也是很重要的。对FlexRecs而言,这需要提供定义及管理推荐工作流的用户界面;对REQUEST而言,这需要提供前端的用户界面及允许用户直观地以交互式的方式明确查询要求。

开发高性能的CARS系统,并在实际应用中检验。大部分情境感知推荐系统的工作还是概念性的,虽然也发展了一些方法,并在一些数据(往往是有限的)上进行了测试,而且在某些指标上表现不错。可CARS领域革命性的进展几乎没有,比如,在设计新颖的数据结构、高效的存储方法以及新的系统架构方面。Hussein等的论文[38]在这方面做了一些工作,作者介绍了一个面向服务的架构,允许为情境感知推荐系统定义和实现了各种不同的“积木”,如推荐算法、情境探测器、各种过滤器和转换器,体现了模块式的风格。这些“积木”可以以各种形式组合和重用 to 系统里进行情境的推荐。这类工作的另一个例子是Abbar等[1],作者用一种面向服务的方法实现了其之前提出的个性化访问模型(PAM),该实现基于作者开发的CARS的全局软件架构[1]。这两篇论文在如何使CARS更加人性化、更具扩展性以及更好的性能的方向上迈出了第一步,后续还需要更多的工作来实现这一目标。

## 7.6 总结

在本章中,我们认为一些情境信息在推荐系统里是有意义的,而且在进行推荐的时候考虑情境信息也是很重要的。我们还阐明了情境信息可以在推荐过程的不同阶段中使用,包括在预过滤及后过滤阶段,也可以作为情境建模中不可分割的一部分。我们已展示了各种使用情境信息的技术,包括上述三种方法,可以组合成一个单一的推荐方法,并对一个案例进行了研究,描述了一种可能的结合方式。

总体而言,情境感知推荐系统(CARS)是相对较新,且正在研究的领域,还有很多工作需要全面开展。本书建议了几种未来的研究方向。总之,CARS包含许多有趣的且有重要实践意义的研究型问题,正成为一个新兴的且有前途的研究领域。

## 致谢

G. Adomavicius的研究部分得到美国国家科学基金会的支持 IIS-0546443。作者感谢 YoungOk Kwon 编辑提供的援助。

## 参考文献

1. Abbar, S., Bouzeghoub, M., and Lopez, S., Context-aware recommender systems: A service-oriented approach. *VLDB PersDB Workshop*, 2009.
2. Abowd, G.D., Atkeson, C.G., Hong, J., Long, S., Kooper, R., and Pinkerton, M., Cyberguide: A mobile context-aware tour guide. *Wireless Networks*, 3(5):421–433, 1997.
3. Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A., Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.
4. Adomavicius, G., and Tuzhilin, A., Incorporating context into recommender systems using multidimensional rating estimation methods. In *Proceedings of the 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces (WPR-SIUI 2005)*, 2005.
5. Adomavicius, G., and Tuzhilin, A., Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
6. Adomavicius, G., Tuzhilin, A., and Zheng, R., REQUEST: A query language for customizing recommendations. *Information System Research*, 22(11)s, 2011.
7. Agrawal, R., Rantzau, R., and Terzi, E., Context-sensitive ranking. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 383–394. ACM, 2006.
8. Ahn, H., Kim, K., and Han, I., Mobile advertisement recommender system using collaborative filtering: MAR-CF. In *Proceedings of the 2006 Conference of the Korea Society of Management Information Systems*, pages 709–715.
9. Akrivas, G., Wallace, M., Andreou, G., Stamou, G., and Kollias, S., Context-sensitive semantic query expansion. In *Proceedings of the IEEE international conference on artificial intelligence systems (ICAIS)*, pages 109–114. Divnomorskoe, Russia, 2002.
10. Alpaydin, E., *Introduction to machine learning*. The MIT Press, 2004.
11. Anand, S.S., and Mopasher, B., Contextual recommendation. *WebMine, LNAI*, 4737:142–160, 2007.
12. Ansari, A., Essegai, S., and Kohli, R., Internet recommendation systems. *Journal of Marketing Research*, 37(3):363–375, 2000.
13. Arbel, R. and Rokach, L., Classifier evaluation under limited resources, *Pattern Recognition Letters*, 27(14): 1619–1631, 2006.
14. Ardissono, L., Goy, A., Petrone, G., Segnan, M., and Torasso, P., Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence*, 17(8):687–714, 2003.
15. Baltrunas, L., and Amatriain, X., Towards time-dependant recommendation based on implicit feedback. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York, 2009.
16. Baltrunas, L., and Ricci, F., Context-dependent items generation in collaborative filtering. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York, 2009.
17. Bazire, M., and P. Brezillon. Understanding context before using it. In Dey, A., and et al., editors, *Proceedings of the 5th International Conference on Modeling and Using Context*. Springer-Verlag, 2005.
18. Berry, M.J., and Linoff, G., *Data mining techniques: for marketing, sales, and customer support*. John Wiley & Sons, Inc. New York, NY, USA, 1997.
19. Bettman, J.R., Luce, M.F., and Payne, J.W., Consumer decision making: A constructive perspective. In M. Tedeschi (editor), *Consumer Behavior and Decision Theory* pages 1–42, 1991.
20. Boutemedjet, S., and Ziou, D., A graphical model for context-aware visual content recommendation. *IEEE Transactions on Multimedia*, 10(1):52–62, 2008.
21. Breese, J.S., Heckerman, D., and Kadie, C., Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, volume 461, pages 43–52. San Francisco, CA, 1998.
22. Brown, B., Chalmers, M., Bell, M., Hall, M., I. MacColl, and Rudman, P., Sharing the square: collaborative leisure in the city streets. In Gellersen, H., Schmidt, K., M. Beaudouin-Lafon, and Mackay, W.E., editors, *Proceedings of the ninth conference on European Conference on Computer Supported Cooperative Work*, pages 427–447. Springer, 2005.
23. Brown, P.J., Bovey, J.D., and Chen, X., Context-aware applications: from the laboratory to

- the marketplace. *IEEE Personal Communications*, 4:58–64, 1997.
24. Burke, R., Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.
  25. Burke, R., Hybrid web recommender systems. *The Adaptive Web*, pages 377–408, 2007.
  26. Cantador, I., and Castells, P., Semantic contextualisation in a news recommender system. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York, 2009.
  27. Cena, F., Console, L., Gena, C., Goy, A., Levi, G., Modeo, S., and Torre, I., Integrating heterogeneous adaptation techniques to build a flexible and usable mobile tourist guide. *AI Communications*, 19(4):369–384, 2006.
  28. Chatterjee, S., Hadi, A.S., and Price, B., *Regression analysis by example*. John Wiley and Sons, 2000.
  29. Chaudhuri, S., and Dayal, U., An overview of data warehousing and olap technology. *ACM SIGMOD Record*, 26(1):65–74, 1997.
  30. Chen, G., and Kotz, D., A survey of context-aware mobile computing research. *Technical Report TR2000-381, Dartmouth Computer Science*, 2000.
  31. Cheverst, K., Davies, N., Mitchell, K., Friday, A., and Efstratiou, C., Developing a context-aware electronic tourist guide: some issues and experiences. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 17–24. ACM, 2000.
  32. De Carolis, B., Mazzotta, I., Novielli, N., and Silvestri, V., Using common sense in providing personalized recommendations in the tourism domain. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York, 2009.
  33. Dey, A.K., Abowd, G.D., and Salber, D., A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16(2):97–166, 2001.
  34. Dourish, P., What we talk about when we talk about context. *Personal and ubiquitous computing*, 8(1):19–30, 2004.
  35. Franklin, D., and Flachsbar, J., All gadget and no representation makes jack a dull environment. In *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments*, pages 155–160. AAAI Press, 1998.
  36. Herlocker, J.L., and Konstan, J.A., Content-independent task-focused recommendation. *IEEE Internet Computing*, pages 40–47, 2001.
  37. Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.T., Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.
  38. Hussein, T., Linder, T., Gaulke, W., and Ziegler, J., Context-aware recommendations on rails. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*, New York, 2009.
  39. Jiang, T., and Tuzhilin, A., Improving personalization solutions through optimal segmentation of customer bases. *IEEE Transactions on Knowledge and Data Engineering*, 21(3):305–320, 2009.
  40. Jones, G., J.F., Glasnevin, D., and Gareth, I., Challenges and opportunities of context-aware information access. In *International Workshop on Ubiquitous Data Management*, pages 53–62, 2005.
  41. Kimball, R., and Ross, M., *The data warehousing toolkit*. John Wiley & Sons, New York, 1996.
  42. Koller, D., and Sahami, M., Toward optimal feature selection. In *Proceedings of the 13th International Conference on Machine Learning*, pages 284–292. Morgan Kaufmann, 1996.
  43. Koren, Y., Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, New York, NY, 2008.
  44. Koutrika, G., Bercovitz, B., and H. Garcia-Molina. Flexrecs: expressing and combining flexible recommendations. In *Proceedings of the 35th SIGMOD international conference on Management of data*, pages 745–758. ACM, Providence, RI, 2009.
  45. Lawrence, S., Context in web search. *IEEE Data Engin. Bulletin*, 23(3):25, 2000.
  46. Leake, D.B., and Scherle, R., Towards context-based search engine selection. In *Proceedings of the 6th international conference on Intelligent user interfaces*, pages 109–112. ACM New York, NY, USA, 2001.
  47. Lilien, G.L., Kotler, P., and Moorthy, K.S., *Marketing models*. Prentice Hall, 1992.
  48. Liu, H., and Motoda, H., *Feature selection for knowledge discovery and data mining*. Springer, 1998.
  49. Lombardi, S., Anand, S.S., and Gorgoglione, M., Context and customer behavior in recommendation. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York.
  50. Lussier, D.A., and Olshavsky, R.W., Task complexity and contingent processing in brand



- choice. *Journal of Consumer Research*, pages 154–165, 1979.
51. Maamar, Z., Benslimane, D., and Narendra, N.C., What can context do for web services? *Communications of the ACM*, 49(12):98–103, 2006.
52. Mokbel, M.F., and Levandoski, J.J., Toward context and preference-aware location-based services. In *Proceedings of the Eighth ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 25–32. ACM, 2009.
53. Oku, K., Nakajima, S., Miyazaki, J., and Uemura, S., Context-aware SVM for context-dependent information recommendation. In *Proceedings of the 7th International Conference on Mobile Data Management*, page 109, 2006.
54. Palmisano, C., Tuzhilin, A., and Gorgoglione, M., Using context to improve predictive modeling of customers in personalization applications. *IEEE Transactions on Knowledge and Data Engineering*, 20(11):1535–1549, 2008.
55. Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., and Pedone, A., Experimental comparison of pre-vs. post-filtering approaches in context-aware recommender systems. In *Proceedings of the 3rd ACM conference on Recommender systems*, pages 265–268. ACM, 2009.
56. Pennock, D.M., and Horvitz, E., Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach. In *IJCAI'99 Workshop: Machine Learning for Information Filtering*, 1999.
57. Prahalad, C.K., Beyond CRM: CK Prahalad predicts customer context is the next big thing. *American Management Association McWorld*, 2004.
58. Ramakrishnan, R., and Gehrke, J., *Database Management Systems*. USA: McGraw Hill Companies, 2000.
59. Ricci, F., and Nguyen, Q.N., Mobyrek: A conversational recommender system for on-the-move travelers. In Fesenmaier, D.R., Werthner, H. and Wober, K.W. (eds.) *Destination Recommendation Systems: Behavioural Foundations and Applications*, CABI Publishing, pages 281–294, 2006.
60. Rodden, T., Cheverst, K., Davies, K., and Dix, A., Exploiting context in HCI design for mobile systems. In *Workshop on Human Computer Interaction with Mobile Devices*, pages 21–22, 1998.
61. Ryan, N., Pascoe, J., and Morse, D., Enhanced Reality Fieldwork: the Context-Aware Archaeological Assistant. Gaffney, V., van Leusen, M., Exxon, S.(eds.) *Computer Applications in Archaeology*. British Archaeological Reports, Oxford, 1997.
62. Sarwar, B., Karypis, G., Konstan, J., and Reidl, J., Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.
63. Schilit, B.N., and Theimer, M.M., Disseminating active map information to mobile hosts. *IEEE network*, 8(5):22–32, 1994.
64. Schiller, J.H., and Voisard, A., *Location-based services*. Morgan Kaufmann, 2004.
65. Sieg, A., Mobasher, B., and Burke, R., Representing context in web search with ontological user profiles. In *Proceedings of the 6th International Conference on Modeling and Using Context*, 2007.
66. Stefanidis, K., Pitoura, E., and Vassiliadis, P., A context-aware preference database system. *International Journal of Pervasive Computing and Communications*, 3(4):439–600, 2007.
67. Umyarov, A., and Tuzhilin, A., Using external aggregate ratings for improving individual recommendations. *ACM Transactions on the Web*, 2010.
68. van Setten, M., Pokraev, S., and Koolwaaij, J., Context-aware recommendations in the mobile tourist application compass. In Nejdl, W., and De, P., Bra, editors, *Adaptive Hypermedia*, pages 235–244. Springer Verlag, 2004.
69. Ward, A., Jones, A., and Hopper, A., A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, 1997.
70. Webster, N., *Webster's new twentieth century dictionary of the English language*. Springfield, MA: Merriam-Webster, Inc., 1980.
71. Woerndl, W., Schueller, C., and Wojtech, R., A hybrid recommender system for context-aware recommendations of mobile applications. In *Proceedings of the 3rd International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces*, pages 871–878, 2007.
72. Yu, Z., Zhou, X., Zhang, D., Chin, C.Y., Wang, X., and Men, J., Supporting context-aware media recommendations for smart phones. *IEEE Pervasive Computing*, 5(3):68–75, 2006.
73. Ziegler, C.N., S. M. McNee, Konstan, J.A., and Lausen, G., Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32. Chiba, Japan, 2005.

# 推荐系统的应用与评估

- 第 8 章 推荐系统评估
- 第 9 章 IPTV 服务提供商推荐系统：一个大规模  
真实产品环境的应用
- 第 10 章 走出实验室的推荐系统
- 第 11 章 匹配推荐系统的技术与领域
- 第 12 章 用于技术强化学习的推荐系统

## 推荐系统评估

Guy Shani 和 Asela Gunawardana

**摘要** 推荐系统目前在商用和研究领域都很流行，在这些领域中已经有很多提供推荐的方法。大部分情况下，如果一个系统设计师想要使用推荐系统，则必须在一组候选方法中做出选择。选择合适算法的第一步是在选择时决定着重于应用程序的哪些属性。的确，推荐系统包含许多可能会影响用户体验的属性，如准确性、健壮性、可扩展性等。本章将讨论如何根据应用程序相关的属性来比较推荐系统。本章将对比研究作为重点，通过某些评估指标对比算法，而不是通过算法的绝对基准。本章将描述合适于算法选择的实验设置。同时考察三种类型的实验：首先进行离线设置的实验，在这个过程中比较推荐技术时无须用户交互；然后是用户调查实验，即通过一组受试对象使用系统并对使用系统的体验做出报告；最后，本章叙述了大规模的在线实验，在实验中使用真正的用户群体与系统交互。在以上的每种情况中本章将叙述不同类型的可以解答的问题，并提供实验方案。还描述了如何从实验中得出可信的结论。然后将仔细研究大量的属性，并根据给定的属性解释如何评估系统。本书还根据评估的上下文属性概述了大量的评估指标。

### 8.1 简介

现在很多应用里都有推荐系统，这些应用为用户展示大量的物品。这些系统一般会为用户提供一个他们可能喜欢的物品的推荐列表，或者预测用户对每个物品的偏好程度。同时帮助用户决定哪些物品适合用户，减轻用户从物品集合中查找偏好物品的工作量。

例如，DVD 租赁商 Netflix<sup>①</sup> 通过显示在列电影的预测评分来帮助用户决定租哪部电影。在线图书零售商 Amazon<sup>②</sup> 提供在列图书的平均用户评分，以及一个购买过某本特定书籍的用户购买的其他书籍的列表。微软公司为用户提供了很多免费下载，如 bug 修复、产品等。当用户下载某个软件时，系统会给出一个补充软件列表一起下载。所有这些系统通常被归类为推荐系统，尽管它们提供的服务不同。

在过去十年里，推荐系统领域已经有了大量的研究，大部分专注于设计新的推荐算法。一个应用程序设计者想要为应用添加推荐系统，那么他将有各种各样的算法可以使用，而且必须决定最适合目标的算法。通常这样的决定是基于比较多个推荐系统性能的实验来实现的。然后，在给定的结构约束上，如类型、实时性、有效数据的可靠性、可用内存以及 CPU 占用率等，设计者可以选出性能最好的算法。另外，大部分提出新算法的研究人员都会将自己的新算法与现有的一系列算法进行比较。通常通过应用提供了候选算法

---

Guy Shani, Microsoft Research, OneMicrosoftWay, Redmond, WA e-mail: guyshani@microsoft.com

Asela Gunawardana, Microsoft Research, One Microsoft Way, Redmond, WA e-mail: aselag@microsoft.com

翻译：北京大学-王雪丽，北京邮电大学-刘树栋 审核：吉林大学-丁彬钊，李曙光，郑州大学-吴宾

① [www.Netflix.com](http://www.Netflix.com)

② [www.amazon.com](http://www.amazon.com)

排名(通常为数值分数)的评估指标进行这类评估。

起初,大多数推荐系统基于准确率(准确预测用户行为的能力)进行评估和排名。然而现在人们普遍认为准确率固然重要,但还不足够部署一个好的推荐系统。在许多应用中人们使用推荐系统不仅是为了预测用户品味的准确率。同时用户也可能对发现新物品、迅速发掘多样性物品、隐私保护、系统响应速度以及其他更多的推荐引擎的属性感兴趣。所以我们必须识别出在具体应用上下文中哪些属性会影响推荐系统,然后才能评估系统在相关属性上的运行性能。

本章仔细研究了推荐系统的评估过程,讨论了三种不同类型的实验:离线实验、用户调查、在线实验。

通常离线实验实施最简单,该实验使用现有的数据和通过对用户行为建模来评估推荐系统性能,如预测准确性。代价较高的选择是用户调查,其中一小组用户被要求使用系统完成一系列任务,通常是事后回答一些关于用户体验的问题。最后,我们就可以在已经部署的系统上做大量的实验,称为在线实验。这些实验基于未意识到实验正在进行的真实用户对系统进行评估。本章讨论了每种实验中可以评估的部分以及不可以评估的部分。

我们有时候能够评估推荐系统完成总体目标的情况。例如,可以在电子商务网站使用和没有使用推荐系统的情况下对比其收入,然后估算推荐系统对该网站的价值。在其他情况下,在特定属性条件下评估推荐系统也是很有用的,这让我们专注于改善那些具有不足之处的属性。首先,必须证明该属性确实与用户相关且影响着用户体验。然后,我们才可以设计算法来改进这些属性。在改进某个属性时,为了属性性能的平衡,有可能会降低其他属性的性能。在许多场景中,我们很难说这些平衡是如何影响系统的整体性能的,这时不得不进行额外的实验来验证或者使用领域专家的观点。

本章重点在于推荐系统面向属性的评估。我们对大量的可能影响推荐系统成败的属性进行了综述,解释了如何使用这些属性对候选推荐进行排序。对每个属性进行三种相关实验:离线实验、用户调查、在线实验。并解释在每种场景下如何进行评估。同时还解释了在评估每个属性时遇到的困难并概述了缺陷。假设更好的处理某个属性能够改善用户体验,对于所有这些属性,我们专注于在该属性上的推荐排名。

本章还回顾以前评估推荐系统的建议,这些建议描述了大量的流行方法,并把这些方法用在评估这些属性的上下文中。我们专注于被广泛研究的准确性和评估方法中的排序指标,并且描述了这些属性的大量的评价指标。对于那些研究比较少的属性,我们提出了方法的来源,并在适当的地方提供了这些方法具体实现的例子。

本章剩下的部分结构如下,在8.2节中将讨论可以评估推荐系统的不同实验设置以及对离线实验、用户调查和在线实验的适当使用。同时还概括了基于这些实验做出可靠决定所需要考虑的问题,包括结果的泛化及统计显著性。在8.3节,我们描述了大量的可能影响推荐系统性能的属性,以及这些属性的评价指标。最后,8.4节进行总结。

## 8.2 实验设置

本节描述了可以用来对推荐进行对比的三个实验阶段。下面的讨论受到了相关领域评估方案的影响,如机器学习、信息检索并强调那些和推荐系统有关的部分。详细内容读者可参阅关于以上领域的文章[49, 13, 60]。

我们首先从离线实验设置开始,通常离线实验最容易执行,因为不需要与用户交互。然后描述用户调查实验,让一组受试对象在受控环境下使用系统,然后做出体验报告。在该实

验阶段,我们能得到该系统的定性和定量的信息,但是在该实验设计阶段一定要考虑各种各样的偏差。最终,也许最可靠的实验便是真实用户使用该系统,特别还是在不知情的情况下。虽然在该实验过程中,我们只能收集到某几种数据,但是实验设置是最贴近现实世界的。

在所有实验情景中,遵循如下实验研究的几条基本规则是很重要的:

- **假设:**在执行实验之前,必须形成一个假设。该假设必须是精确和严密的,设计实验来验证这个假设也是很重要的。例如,假设算法 A 预测用户评分比算法 B 预测的更好,在该情况下,就应该测试预测准确率而不是其他因素。
- **变量控制:**当基于某个假设来比较候选算法时,将所有不被测试的变量设为固定值是很重要的。举例说明,如果我们要比较算法 A 和算法 B 在电影评分上的准确率,两者使用不同的协同过滤模型。如果我们使用 MovieLens 数据集训练算法 A,使用 Netflix 数据集训练算法 B,而且算法 A 性能优越,但是我们无法判断这是由于 CF 模型的优越性,还是因为输入比较好的数据,或者两者都是。因此我们必须在相同数据集上训练算法(或者来自于相同数据集的无偏差样本),或者为了了解性能优越的原因,在两个不同数据集上训练同一个算法。
- **泛化能力:**在做实验总结时,我们可能希望结论能够不仅概括当前的实验。当为真实应用选择算法时,我们可能希望自己的结论在已部署的系统中得到保持,并且泛化超出我们的实验数据。同样,当开发新算法时,我们希望自己的结论在实验使用的特定应用和数据集之外得到保持。为了增加结果泛化的可能性,通常我们必须尝试几个数据集合应用程序。了解所使用的各种各样的数据集的属性是很重要的。总体来说,使用的数据越多样化,结果就越泛化。

## 8.2.1 离线实验

离线实验通过预先收集用户选择的或者打过分的物品的数据集来进行实验。使用该数据集我们可以尝试模拟用户与推荐系统的交互行为。在这种情况下,假设在收集数据时的用户行为与推荐系统部署后的用户行为是足够相似的,因此我们可以基于模拟做出可靠的决策。离线实验是很有吸引力的,因为不需要真实用户的参与,这样可以在低成本的情况下比较大量的算法。离线实验的不足之处是能够回答问题的范围较小,具有代表性的是关于算法预测性能的问题。特别是,我们必须假设与系统进行交互的用户行为已经通过在系统部署之前的用户行为进行了恰当的模拟,其中包括推荐系统的选择。因此,不能在此阶段直接评估推荐对用户行为的影响。

因此,离线实验的目的是过滤不恰当的方法,为成本比较大的用户调查和在线实验提供相对较小的算法候选集进行测试。在此过程中的一个典型例子是在离线实验阶段调优算法参数,这样在进行下个阶段时,算法就具有最优的参数。

### 8.2.1.1 离线实验数据集

由于离线实验的目的是过滤算法,那么离线评估所使用的数据应该与设计者期望推荐系统在线部署后所面临的数据尽可能地匹配。而且必须确保所选的用户、物品和评分的分布没有偏置。例如,在这样的情况下,数据从现有的推荐系统(或者一个没有推荐功能的系统)中获得,实验设置者为了降低实验成本,可能会尝试通过剔除计数较低的用户和物品对数据进行预过滤。如果这样做,实验者应该注意到这涉及一个权衡,因为这为数据引进了系统性偏差。如果必要,对用户和物品进行随机取样是一个减少数据的不错的方法,



虽然这也可能引进别的偏差到实验中(例如,这可能会对能够很好处理稀疏数据的算法更有利)。有时,已知的数据偏差可以通过如重新加权数据来纠正,但纠正数据中的偏差是件很困难的事情。

另一个偏差的来源可能来自于数据集本身。例如,某些用户可能比较喜欢给自己有强烈观点的物品打分,有些用户的评分比其他用户多。因此,有明确评分的项目集可能因评分本身使其具有偏差[38]。再一次,如对测试数据进行重新取样或重新加权可以用来尝试纠正这些偏差。

### 8.2.1.2 模拟用户行为

为了离线评估算法,必须模拟系统预测和推荐、用户纠正预测和使用推荐的在线过程。该过程通常通过记录用户历史数据,然后隐藏某些交互来模拟用户如何为一个物品打分,或者用户会单击哪些推荐来实现。有很多方法可以用来选择被评分或被选择的物品进行隐藏。再次,在做选择时比较可取的方法是尽可能接近地模拟实际目标。但是,在许多场景中,我们受到评估方法计算成本的限制,而且为了在大型数据集上进行实验必须做出妥协。

理想情况下,假设系统在收集数据时一直是运行的,如果我们能够访问用户选择物品的时间戳,那么我们就可以模拟推荐系统的预测。开始时我们没有可用的先验数据来计算预测,可以通过用户选择的时间顺序和步骤去尝试预测每一个选择,并使得该预测可用在将来的预测中。对于大型数据集,一个简单的方法是随机抽样,对用户操作之前的时间进行抽样,在用户操作之后隐藏所有选择(所有用户的选择),然后尝试推荐物品给该用户。该方法需要更改推荐前的数据,计算开销是相当昂贵的。

一个代价比较小的选择是对测试用户进行抽样,然后抽取单一的测试时间,在测试时间过后对每个测试用户隐藏所有物品。这个过程模拟了这样一个场景,推荐系统是在测试时建立的,而且在进行推荐时不考虑测试结束后获得的新数据。另一个选择就是为每个测试用户抽样一个测试时间,测试时间过后隐藏测试用户的物品,不保持用户的时间一致性。这种选择实际上假设物品选择的顺序是重要的而不是做出选择的绝对时间。最后一个选择就是忽略时间。我们先抽样一组测试用户,然后抽样选取对每个用户  $a$  隐藏的物品数目  $n_a$ ,最后抽样  $n_a$  个需要隐藏的物品。这是假设用户选择的时间因素是不重要的。如果用户操作的时间戳是未知的,我们只能强行这种假设。后面的三种方法,将数据分成了单一的训练数据和单一的测试数据。在给定的约束下,为领域和感兴趣的任務(请参照第11章)选择一个最合适而不是最方便的方法。

在很多研究论文中常用的方法是,使用固定数目的已知物品以及每个测试用户的固定数目的隐藏物品(所谓的“给定  $n$ ”或者“除去  $n$  个之外所有的”方法)。这个方案在分析判断算法并确定在哪些情况下算法表现好时很有用。然而,当我们想要决定在我们的应用程序中使用哪种算法时,必须扪心自问,是否真的只为那些正好已经标记了  $n$  个物品的用户展示推荐还是那些预计继续给  $n$  个商品打分的用户。如果并不是这样的情况,那么用这些方法计算出来的结果是有偏差的,使得预测算法在线行为不可靠。

### 8.2.1.3 更复杂的用户模型

以上我们所讨论的所有方法对用户的行为进行了一些假设,这些行为在某些特定的应用程序中可以作为用户模型。虽然只讨论了很简单的用户模型,但有时可能需要更复杂些的用户行为模型[37]。采用先进的用户模型可以模拟用户与系统的交互,从而减少昂贵的用户调查和在线实验的开销。然而在设计用户模型时必须注意:第一,用户建模很难,在

这方面业界有大量的研究(如文献[15])。第二,当用户模型不准确时,我们可能在优化一个模拟性能与实际性能毫不相关的系统。虽然设计一个使用复杂用户模型的算法进行推荐是合理的,但是对使用这样复杂难以验证的用户模型来验证的实验要谨慎。

### 8.2.2 用户调查

许多推荐方法依赖于用户和系统的交互(参照第23、13、6章)。模拟一个可靠的用户与系统交互很难,因此离线测试很难执行。为了合理地评估这种系统,必须收集真实用户与系统的交互。一旦当离线测试是可行的,真实用户的交互还可以提供关于系统性能的其他信息。在这种情况下,我们通常会进行用户调查。

用户调查是由招聘一组测试对象,并要求他们执行一些需要与系统进行交互的任务来进行的。在受试对象执行任务时,我们观察记录他们的行为,收集任意数量的量评测方法,例如,完成了任务的哪些部分,任务结果的准确性,或者执行任务的时间。在任务完成之前、期间、之后的情况下我们可以提出许多定性问题。这些问题可以帮助收集一些不能直接观测到的数据,例如,受试对象是否喜欢用户界面,或者用户觉得完成任务是否轻松。一个典型的实验例子是测试推荐算法对新闻故事浏览行为的影响。在该例子中,受试对象阅读一组他们感兴趣的故事,这些故事可能包含相关的推荐也可能不包含。然后我们可以确认推荐是否被使用,有推荐和无推荐时人们阅读的故事是否不同。可以收集数据,例如,某个推荐的单击次数,甚至在某些情况下,追踪眼球运动来观察受试对象是否看过该推荐。最后,我们可以提出一些定性问题,如受试对象是否认为这些推荐是相关的。

当然,在许多其他领域用户调查是核心工具,因此在用户调查的合理设计方面有很多文献。该章节只概览了通过用户调查评估推荐系统时应该考虑的基本条件,感兴趣的读者可以在其他章节找到更深入的讨论(如文献[5])。

#### 8.2.2.1 优点和缺点

用户调查或许能回答我们研究的三个实验设置的大部分问题。不同于离线实验,该设置允许我们测试用户与系统交互时的行为,以及推荐对用户行为的影响。在离线情况下的用户调查测试中,通常会做出如“给定相关的推荐,用户很可能会使用”这样的假设。同样,这是唯一一个设置允许我们收集定性数据,这对于解释量化结果是至关重要的。

同时,我们通常可以在该设置中收集大量的量化方法,因为可以近距离地监视用户执行任务。然而用户调查也有一些缺陷。首先,用户调查执行代价昂贵,招募大量的受试对象并要求他们执行足够量的任务,要么会花费大量的时间,如果受试对象是志愿者;要么会开销很大,如果受试对象是付费雇佣。因此,通常我们必须限制在一小组受试对象及一组相对较小的任务集,而且不能测试所有可能的情况。此外,为得到可靠的结果,每种情况都必须重复几次,这就更进一步地限制了可测试的不同任务的范围。

由于这些实验执行代价昂贵,我们应该在尽可能低的粒度下收集尽可能多的数据。这将允许我们以后详细地研究实验结果,分析实验之前不明显的条件。该指南可以帮助我们减少为收集忽略的评测方法所需的连续实验。

另外,为了避免实验失败,例如,由于某些用户行为导致程序出错,研究人员通常会进行试点用户调查。这种实验都是小型实验,不是为了收集统计数据,而是为了测试系统漏洞和故障。在某些情况下,这些试点研究结果被用来改善推荐。如果是这样,那么试点结果成为“污点”,在最终用户调查计算评测方法时不应使用。

另一个重要的考虑是测试对象必须尽可能地贴近真实系统的用户群。例如,如果系统是为推荐电影而设计的,那么在狂热影迷上的研究结果或许并不能覆盖整个人口。该问题经常出现在研究参与者是自愿者,因为在此情况下,原本对应用更感兴趣的人会更容易成为志愿者。

然而,即使受试对象正好代表了真正的用户群,结果依然可能有偏差,因为他们被告知在参加一个实验。例如,众所周知付费的受试对象会倾向于尝试迎合执行实验的人或公司。如果受试对象知道要测试的假设,他们可能会下意识地提供支持该假设的证据。为了调节这一点,在收集数据之前最好不要透露实验目的。另外,受试对象的报酬根据他们选择的物品采取完全或部分补贴时,出现了另外一个更加微妙的效果。在系统最终用户的补贴不同的情况下,数据可能出现偏差,因为当付全价时用户的选择和偏好可能不同。

#### 8.2.2.2 受试对象间与受试对象内

由于通常用户调查要比较几个候选方法,每个候选方法必须在相同任务上进行测试。为测试所有候选方法,我们可以在受试对象间比较候选方法,将每个受试对象分配到一个候选方法进行实验,或者在受试对象内比较候选方法,每个受试对象在不同的任务上测试一组候选方法[20]。

通常受试对象内实验可提供更多的信息,因为在候选方法之间进行有偏差的用户分割不能说明一个方法的优越性。在该设置中可能会提出关于不同候选方法的比较问题,比如,受试对象更喜欢哪个候选方法。然而,这类测试用户更能意识到实验的存在,使得隐藏候选方法间的不同更加困难。

受试对象间实验,也称为 A-B 测试(All Between),提供的设置更贴近真实系统,即因为每个用户实验都有一个单独的系统。这样的实验也可以测试使用系统的长期效果,因为用户无须更换系统。因此我们可以检验用户是如何适应系统的,然后估计一个专业的学习曲线。

#### 8.2.2.3 变量调节

正如上面所述,控制所有没有进行专门测试的变量是重要的。然而,要将几个候选方法的输出显示给受试者,正如受试对象内实验一样,我们必须调节几个变量。

为受试对象显示结果时,可以是顺序地或者混合地。两种情况中都有某些偏差需要校正。将结果顺序显示时,之前看到的结果会影响用户对现在结果的看法。例如,如果显示之前的结果看起来不合适,那么后面显示的结果就可能看起来要比实际上好。在展示两组结果时,由于位置不同可能会产生某些偏差。例如,来自许多文化背景的用户趋向于从左至右、从上到下地查看结果。因此,用户可能优先看到显示在上面的结果。

校正这种未测试变量的一个常用方法是拉丁方[5]过程。该程序每次将各种结果的位置随机化,从而抵消了由未测试变量带来的偏差。

#### 8.2.2.4 调查问卷

用户调查允许我们使用强大的调查问卷工具。在受试对象执行任务之前、期间和之后我们都可以问他们一些体验问题。这些问题可以提供一些很难评测的属性的信息,如受试对象的心理状态,或者受试对象是否喜欢该系统。

这些问题在提供有价值信息的同时也会提供误导信息。提出不需要“正确”答案的中立问题是很重要的。受试对象回答也可能不真实,比如,他们认为答案是隐私的,或者他们认为真实的答案会让他们不讨人喜欢。

其实,在其他的领域有大量的关于调查问卷编写艺术的研究,详细内容读者可以参考文献[46]。

### 8.2.3 在线评估

在许多实际推荐应用程序中,系统设计者希望可以影响用户的行为。因此我们对评测用户与不同推荐系统互动时的行为变化感兴趣。例如,在其他条件相同的情况下,如果一个系统的用户经常接纳推荐,或者在同等条件下从一个系统用户收集的效用超过从另一个系统用户收集的效用,那么我们就可以说一个系统优于另一个系统。

推荐系统的实际效果取决于许多因素,例如,用户的意图(他们需要何种具体的信息,他们在寻求多大的新颖性与多大的风险),用户上下文(他们已经熟悉哪些物品,他们有多信任该系统),推荐显示的界面。

因此,为系统真实价值提供强有力证据的实验是在线评估,期间真实用户使用系统执行真实的任务。对比几个在线系统是最可靠的,获取不同系统的排名,而不是难以解释的绝对数字。

出于该原因,许多实际系统采用能够比较多个算法的在线测试系统[32]。通常,这种系统将一小部分浏览重定向到不同的推荐引擎,然后记录用户与不同系统之间的交互。

进行这种测试时必须考虑几点因素。例如,对用户随机抽样(重定向)是很重要的,只有这样方法的比较才公平。选出推荐的不同方面同样重要。例如,如果我们关心的是算法的准确度,那么用户界面保持一致是很重要的。相反如果我们把重点放在更好的用户界面上,那么底层算法最好保持一致。

在某些情况下,这样的实验是有风险的。例如,一个提供不相关推荐的测试系统可能会阻止测试用户再次使用真实系统。因此,实验对系统会有负面影响,这可能是商业应用程序无法接受的。

由于上述原因,在线评估测试最好运行在证明候选方法是合理的大量离线研究或者评测用户对系统态度的用户调查之后。这个循序渐进的过程减少了引起用户严重不满的风险。

在线评估的独特之处在于允许对系统目标进行直接评测。例如,长期利润和用户留存。因此,可以用来推断系统属性是如何影响整体目标的,例如,推荐准确性、多样性,还可以用来理解属性之间的平衡。然而,由于将这些不同的属性独立是很难的,再加上在线比较算法代价昂贵,很难对这些关系获得一个全面完整的判断。

### 8.2.4 得出可靠结论

在任何类型的实验中,能够确信候选推荐对于系统将来面临的未知数据仍是一个好的选择,这是很重要的。正如我们之前所解释的,为了最佳地模拟在线应用,我们应该谨慎地选择离线实验的数据和用户调查中的受试对象。然而仍然有这样的可能性,在这个测试集上性能好的算法在另一个测试集上性能也很好,因为实验恰巧适合该算法。为减少这种统计错误的可能性,必须对结果进行显著性检验。

#### 8.2.4.1 置信度和 $p$ 值

显著性检验的一个标准工具是显著性水平和  $p$  值——因幸运而获得结果的概率。通常,我们拒绝零假设,即如果  $p$  值超过某一阈值那么算法 A 并不比算法 B 好。也就是如果排名是在超过阈值的情况下观察得到的,那么该实验结果并不重要。习惯上,取  $p =$

0.05 作为阈值,也就是置信度小于 95%。在错误选择的代价较高的情况下可以使用更严格的显著性水平(如 0.01 或者更低)。

#### 8.2.4.2 配对结果

为了显著检验来证明算法 A 确实优于算法 B,我们通常使用几个独立实验的结果来比较算法 A 和算法 B。事实上,我们曾提出的生成测试数据的方法(8.2.1.2 节)能使我们得到这样一组结果。假设测试用户是从一些人中独立抽取的,对每个测试用户进行算法性能评估会使我们得到所需的独立比较。然而,当为同一个用户进行多物品推荐或预测时,结果中单物品性能指标不可能独立。因此,最好以单用户为基础进行算法比较。

对算法 A 和算法 B 进行这种单用户配对性能评测,有一个简单的显著检验方法符号检验[13]。计数算法 A 优于算法 B 的用户数  $B(n_A)$  和算法 B 优于算法 A 的用户数  $A(n_B)$ 。显著性水平即 A 不真正优于 B 的概率,估计至少有  $n_A/(n_A+n_B)0.5$  的概率二项式实验成功(如公平掷硬币有  $n_A/(n_A+n_B)$  概率硬币朝上),由下式给出:

$$p = (0.5)^n \sum_{i=n_A}^n \frac{n!}{i!(n-i)!} \quad (8.1)$$

符号检验的简单性以及缺少对分布情况的假设使得它是个不错的选择。虽然用户数相同通常会被忽略,但 Demšar[13]建议将它们平均分配给算法 A 和算法 B,因为零假设得到验证即算法差异不显著。当  $n_a+n_b$  很大时,我们可以通过正态分布利用大样本理论进行近似式(8.1)。然而,在强大的现代计算机条件下,这通常是不必要的。一些作者(如文献[49])使用名词 McNemar's test 来表示对两边符号检验使用  $\chi^2$  近似。

注意,有时即使系统 B 的性能优于系统 A 的性能,符号检验可能会显示系统 A 优于系统 B 的概率很高。这种情况发生在系统 B 的性能偶尔压倒性优于系统 A。因此,这个看似不一致的结果是由于只测试了一个系统优于另一个系统的概率,而没有考虑差异梯度。

符号检验可以扩展到那些我们想知道一个系统优于另一个系统的概率的值的的情况。例如,假设系统 A 比系统 B 资源更加密集,而且只有在优于系统 B 一定量时才值得部署。我们将符号检验中的“成功”定义为 A 以一定数量优于 B,找到 A 并不真正以该数量优于 B 的概率作为式(8.1)里的  $p$  值。

一个常用的考虑差异梯度的检验是配对学生  $t$  检验,该检验着眼于算法 A 和算法 B 性能分数的平均差异,使用差异分数的标准差进行归一化。使用该检验需要不同用户的分数差异是可比的,这样平均这些差异才是合理的。对于少数用户,检验的有效性还取决于正态分布的差异。Demšar[13]指出当样本量很小时该假设是很难验证的, $t$  检验对异常值敏感。他推荐使用 Wilcoxon 符号秩检验,与  $t$  检验相似,使用算法 A 和 B 间的差异梯度,但在差异上不做分布假设。然而,使用 Wilcoxon 符号秩检验依然需要两个系统的差异在用户间是可比的。

另一个提高结论显著性的方法是使用大数据集。在离线情况下,需要使用较小的训练集,可能导致一个不能代表在部署后可用的训练数据数量的实验性协议。在用户调查中这意味着额外的开销。在线测试中,增加为算法收集的数据需要长时间试用或者少量算法比较的额外开销。

#### 8.2.4.3 非配对结果

上述测试适合观察值是成对的情况。也就是说每个算法在每个测试用例上运行,正如离线测试时一样。而在线测试中,通常用户被分配到一个或者另一个算法上,这样两个算



法就不会在相同测试用例上进行评估。在此情况下 Mann-hitney 检验是 Wilcoxon 检验的一个扩展。假设算法 A 结果为  $n_A$ , 算法 B 结果为  $n_B$ 。

这两个算法的性能评估被汇集排序, 以使得最好的结果排在最前, 最差的排在最后。相同结果的排序被平均。例如, 如果第二至第五名相同, 排名都为 3.5。Mann-Whitney 检验计算了零假设概率, 即从  $n_A + n_B$  中随机选择结果至少要与从算法 A 的结果  $n_A$  选择结果的平均排名一样好。

概率可以通过列举所有  $\frac{(n_A + n_B)!}{n_A! n_B!}$  个选择以及计算可以达到所需平均排名的选择进行精确计算, 或者通过重复  $n_A$  个结果重复取样来近似。当  $n_A$  和  $n_B$  足够大时(一般超过 5), 从满足零假设的  $n_A + n_B$  池中随机选择的  $n_A$  结果的平均排名的分布可使用期望为  $\frac{1}{2}(n_A + n_B + 1)$  以及标准方差为  $\sqrt{\frac{1}{12n_B}(n_A + n_B + 1)}$  的高斯分布进行近似。因此, 我们可以通过减去  $\frac{1}{2}(n_A + n_B + 1)$  再除以  $\sqrt{\frac{1}{12n_B}(n_A + n_B + 1)}$  来计算来自系统 A 的  $n_A$  结果的平均排名, 在该值上评估标准高斯 CDF 并得到该检验的  $p$  值。

#### 8.2.4.4 多重测试

另外一个要考虑的重要因素, 主要是在离线情况下, 即评估算法的多个版本的效果。例如, 实验者可能会尝试一个新的推荐算法的多个变体, 并与基准算法进行比较, 直到找到一个算法通过  $p=0.05$  的符号检验, 从而推断他们的算法在基准算法上提高了 95% 的置信度。然而, 这并不是一个有效的推断。假设实验者评估了 10 个变体, 而他们全部在统计上与基准相同。如果任何一个实验错误地通过了符号检验的概率为  $p=0.05$ , 那么 10 个实验中至少一个实验错误地通过符号检验的概率为  $1 - (1 - 0.05)^{10} = 0.40$ 。该风险俗称“测试集调优”, 可以通过将测试用户分为两部分——一个研究(调优)集和一个评估集来避免。基于研究集选择算法, 选择的有效性通过在评估集上运行显著性检验来评测。

为许多算法进行排序时存在同样的担忧, 但更加难以避免。假设最好的  $N+1$  个算法是在研究集中选择的, 为得到使得选择的算法确实为最好算法的置信度  $1-p$ , 那么它必须在评估集中优于其他  $N$  个算法, 且显著性为  $1 - (1-p)^{1/N}$ 。这就是著名的 Bonferroni 校正法, 应该用在配对显著性检验被多次使用时。另外的方法如方差分析或者弗里德曼检验, 是学生  $t$  检验和 Wilcoxon 符号秩检验的泛化。方差分析对不同算法性能检验的正态性及其差异的关系做出强假设。这里推荐读者阅读文献[13]对以上或其他多重算法排序测试进行深入研究。

当使用多种方法比较一对算法时会产生更加微妙的担忧。例如, 对比两个算法时可能会用到许多准确率评测方法和覆盖率评测方法等。即使两个算法在所有评测下相同, 找到一个评测方法使得一个算法以一定显著水平优于另一个算法的概率会随着评测方法的增加而增加。如果不同的评测方法是独立的, 上述的 Bonferroni 校正法可以使用。然而, 由于评测方法往往是相关的, Bonferroni 校正法可能过于严格, 可以使用别的方法如控制错误发现率[2]。

#### 8.2.4.5 置信区间

尽管我们将注意力集中在对比研究上, 其中必须在一组候选算法中选择最合适的一个, 这有时需要评测一些属性的价值。例如, 管理员可能想估计系统预测的错误, 或者系

统赚取的净利润。当度量这些数量时,了解估算的可靠性是很重要的。一个流行的方法是计算置信区间。

例如,也许估算系统的 RMSE 预期为 1.2,那么落在在 1.1~1.35 的概率为 0.95。计算置信区间的最简单方法是假设这些兴趣量成高斯分布,然后通过多重独立的观察值计算期望和标准方差。当我们有很多观察值时,可以通过非参数方法计算兴趣量的分布,如直方图,以及找到包含具有理想概率的兴趣量的上下界来去除该假设。

### 8.3 推荐系统属性

本节讨论一系列属性,在决定选择哪种方法进行推荐时这些属性需要考虑。由于不同的应用程序有不同的需要,系统设计者必须决定对手头的具体应用的哪些重要属性进行评测。一些属性可能会被折衷,最明显的一个例子或许是当其他属性(如多样性)改善时,准确率降低了。了解并评估这些折衷和它们对整体效果的影响是重要的。然而,如何在无须大量在线测试和领域专家知识的情况下以恰当的方式获得这样的认识还是一个未解决的问题。

另外,其中许多属性对用户体验的影响是未知的,而且是取决于应用程序。虽然我们确实可以推测出用户会喜欢不同的推荐或者被告知置信区间界限,但证明这在实际中确实很重要是必要的。因此,当你提出改善其中一个熟悉的方法时,也应该评估一下该属性的改变如何影响用户体验,可以通过用户调查或者在线实验。

该实验一般会使用单一的推荐方法,该方法有一个可调整参数来影响被考虑在内的属性。例如,我们可以设想一个参数控制推荐列表的多样性。然后,物品应该以基于该参数不同值的推荐进行显示,我们应该评测参数对用户体验的影响。我们不应该评测用户是否注意到了属性的变化,而应该评测属性的变化是否影响了用户与系统的交互。正如在用户调查情况下一样,最好不让用户调查的受试对象和在线实验的用户知道实验目的。很难设想这个过程在离线实验下进行,因为我们需要知道用户对参数的反应。

一旦某个特定的系统属性对手头应用程序的用户的影响被充分理解,那么我们就可以用这些属性的差异来选择推荐。

#### 8.3.1 用户偏好

在本章中我们感兴趣的是选择问题,我们需要从一组候选算法中选择一个算法,一个显而易见的选择就是进行用户调查(对象内),请求参与者选择其中一个系统[25]。这个评估并不把受试对象限制在特定的属性上,通常用户做出这种判断要比给体验打分更容易。然后,我们可以选择得票数最多的系统。

然而,除了之前讨论的用户调查的偏差,我们还应该知道有一些其他的问题。首先,上述方案假设所有用户是平等的,这可能并不总是正确的。比如,一个电子商务网站可能更喜欢购买了很多物品的用户的观点而不是只买了一个物品的用户的观点。因此如果可行,我们需要进一步根据用户的重要性为投票加权重。在用户调查中分配正确的重要性权重可能并不容易。

也有可能出现这种情况,喜欢 A 系统的用户只是稍微喜欢,而喜欢系统 B 的用户可能对 A 的评价很低。在这种情况下,即使有很多的用户选择系统 A,实际可能还是想选择系统 B。为了评测该情况,在做用户调查时,对偏好问题我们需要无二义性的答案。于

是, 用户之间的校准分数问题出现了。

最后, 当我们想改善一个系统时, 了解人们喜欢这个系统超过另一个系统的原因是很重要的。通常, 在比较特定的属性时, 比较容易理解。因此, 用户满意度的预测是重要的, 而将满意度分为较小组件有助于理解并改善系统。

### 8.3.2 预测准确度

预测准确度是目前在推荐系统研究中讨论最多的属性。在大多数推荐系统底层有一个预测引擎。这个引擎可能预测用户对物品的看法(如对电影的打分)或者使用概率(如购买)。

推荐系统中有一个基本的假设, 即用户更喜欢推荐更精确的系统。因此许多研究人员着手于寻找提供更精确预测的算法。

预测准确度通常独立于用户界面, 可以通过离线实验评测。在用户调查中评测预测准确度是评测给定推荐的准确度。这与没有推荐的用户行为预测是不同的概念, 它更接近真实系统中的真实准确度。

我们在这讨论三大类预测准确度评测: 评分预测准确度评测, 使用预测准确度评测, 物品排名预测准确度评测。

#### 8.3.2.1 评分预测准确度评测

在一些应用中, 如在流行的 Netflix DVD 租赁服务的最新发布页面, 我们希望预测用户对某个物品的评分(如 1 到 5 星)。在这种情况下我们希望评测系统评分预测的准确度。

均方根误差(RMSE)大概是在评估评分预测中最流行的度量。系统在已知用户-物品集合 $(u, i)$ 的真实评分 $r_{ui}$ 情况下在测试集 $\mathcal{T}$ 上生成预测评分 $\hat{r}_{ui}$ 。通常 $r_{ui}$ 为已知是因为它们隐藏在离线实验中或者是通过用户调查或在线实验获得的。预测评分与实际评分间的 RMSE 由下式给出:

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} (\hat{r}_{ui} - r_{ui})^2} \quad (8.2)$$

平均绝对误差(MAE)也是一个常用的选择, 由下式给出:

$$\text{MAE} = \sqrt{\frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} |\hat{r}_{ui} - r_{ui}|} \quad (8.3)$$

相比于 MAE, RMSE 不成比例地惩罚大的误差, 因此, 在给定有 4 个隐藏物品的测试集的情况下, RMSE 适用在前三个评分误差为 2、第四个评分误差为 0 的系统, 而 MAE 适用于一个评分误差为 3、其他三个误差为 0 的系统。

归一化均方根误差(NRMSE)和归一化平均绝对误差(NMAE)是在一定评分范围内(如)归一化了的 RMSE 和 MAE。由于它们只是 RMSE 和 MAE 的缩减版本, 算法排序结果与未归一化时相同。

平均 RMSE 和平均 MAE 调整非均衡的测试集。例如, 如果测试集的物品分布不均衡, 在此测试集上计算出的 RMSE 或 MAE 可能会受到高频物品产生的误差的严重影响。如果我们需要一个方法来代表任意物品的预测误差, 那么最好分别计算每个物品的 MAE 或 RMSE, 然后再取所有物品的平均值。类似地, 如果测试集的用户分布不均衡, 又希望推断出一个随机抽取的用户所要面临的预测误差, 那么我们可以计算每个用户的平均 RMSE 和 MAE。

RMSE 和 MAE 仅取决于误差的幅度大小。而在某些应用中, 评分的语义可能使得预测误差不仅取决于其幅度大小。在这样的域中可能需要一个合适的失真测量 $d(\hat{r}, r)$ 而不

是平方差或者绝对差。例如，一个应用有一个 3 星系统，1 表示“不喜欢”，2 表示“中立”，3 表示“喜欢”，而且推荐给用户不喜欢的物品比不给用户推荐物品更糟糕，这时一个设置为  $d(3, 1)=5$ ,  $d(2, 1)=3$ ,  $d(3, 2)=3$ ,  $d(1, 2)=1$ ,  $d(2, 3)=1$  和  $d(1, 3)=2$  的失真测量可能是合理的。

### 8.3.2.2 使用预测准确度评测

在许多应用中，推荐系统并不预测用户对物品的偏好，如电影评分，而是向用户推荐他们可能会用到的物品。例如，当电影被添加到队列时，Netflix 会根据加入的电影推荐一组用户可能感兴趣的电影。在这种情况下，我们感兴趣的不是系统是否正确地预测这些电影的评分，而是系统是否正确地预测用户是否会将电影放入队列(使用物品)。

在对使用预测进行离线评估时，我们通常有一组由每个用户使用过的物品组成的数据集。然后选择一个测试用户，隐藏他的一些选择，并要求推荐预测一组用户会用到的物品集。对于推荐和隐藏的物品，有 4 种可能的结果，如表 8.1 所示。

表 8.1 为用户推荐项目可能结果的分类

	被推荐	未被推荐
使用	真阳性数(tp)	假阴性数(fn)
未被使用	假阳性数(fp)	真阴性数(tn)

在离线情况下，由于数据并不是由经过评估的推荐系统收集的，我们被迫假设没有被使用的物品即使被推荐了也不会被使用。例如，用户不感兴趣或者对用户无用。这个假设有可能不成立，比如，当未被使用的物品包含一些用户没有选择但是感兴趣的物品时。例如，用户没有使用某个物品是因为不知道它的存在，当推荐展示了该物品时，用户就可以决定选择它了。在这种情况下，假阳性数的数量被高估了。

我们可以计算落入表中每个单元格的例子的数量，并计算如下数值：

$$\text{Precision} = \frac{\# \text{tp}}{\# \text{tp} + \# \text{fp}} \quad (8.4)$$

$$\text{Recall(True Positive Rate)} = \frac{\# \text{tp}}{\# \text{tp} + \# \text{fn}} \quad (8.5)$$

$$\text{False Positive Rate}(1 - \text{Specificity}) = \frac{\# \text{fp}}{\# \text{fp} + \# \text{tn}} \quad (8.6)$$

通常我们已知这些数值之间可以互相影响抵消。如果允许更长的推荐列表通常会提高查全率，但也可能会减少查准率。如果应用中展示的推荐数量是预定的，计算兴趣最有用的方法是计算 Top-N 查准率。

在为用户展示的推荐数量未预定的应用中，评估算法时最好在一定推荐列表长度范围内，而不是使用固定的长度。因此，我们可以计算查准率和查全率的比较曲线或者真阴性数和假阴性数的比例曲线。前者被简化成为查准率—查全率曲线，后者被简化成为观测者操作特征<sup>⊖</sup>(Receiver Operating Characteristic)或 ROC 曲线。虽然两种曲线都是为了计算用户偏好物品占真正推荐的物品的百分比，但查准率—查全率曲线着重于用户偏好的物品的比例，而 ROC 则着重于最终被推荐但用户不喜欢的推荐比例。

我们应该根据领域的属性和应用的目标来决定使用查准率—查全率曲线还是 ROC 曲

⊖ 其出处涉及信号检测理论。

线。例如,假设有一个在线视频租赁服务为用户推荐 DVD。它的查准率评测是指真正适合用户的推荐比例。无论不合适推荐所展示的是一小部分还是大部分,曾被推荐过的不合适 DVD 都与系统为用户推荐的相关物品所占比例无关,因此查准率-查全率曲线适用该应用。另一方面,考虑一个选择物品向用户销售的推荐系统,比如,将物品通过邮件发送给用户,如果用户不购买则不需要成本。在这种对实现尽可能多的潜在销售的同时最小化市场成本感兴趣情况下,ROC 曲线要比查准率-查全率曲线更具有相关性。

给定两个算法,我们为每个算法计算出一对这样的曲线,如果一种曲线完全优于另外一种曲线,那么决定优越的算法变得容易。然而,当曲线相交时,决定就不那么明显,取决于讨论中的应用。应用的相关知识将告诉我们决定应基于曲线的哪个区域。

归纳 ROC 和查准率-查全率的方法,如 F 测量[58]及 ROC 曲线下方面积(AUC)[1],对应用中算法的独立比较是有用的,但当为一个特定的任务选择算法时,更好的方法是通过能够反映现有特定需求的方法进行选择。

### 多个用户的查准率-查全率和 ROC 曲线

当评估多个用户的查准率-查全率和 ROC 曲线时,根据不同的应用有很多策略可以用来加强结果。

在为每个用户都制定固定数量推荐结果的应用中(例如,当用户访问一个新闻门户时显示固定数量的新闻头条),我们可以为每个用户计算推荐列表长度为  $N$  时的查准率和查全率(或真阴性率和假阴性率),然后在每个  $N$  上计算正确度和查全率(或真阴性率和假阴性率)的平均值[51]。由此得到的曲线是十分有价值的,因为它们为每个可达到的查准率和查全率规定了一个  $N$  值(或真阴性率和假阴性率),反过来,该曲线还可以用来评估在某一给定  $N$  值上的性能。以这种方式得到的曲线叫作 Customer ROC 曲线(CROC)[52]。

当可以向每个用户展示不同数目的推荐时(如向每个用户展示所有被推荐电影的集合),我们可以通过以下过程计算 ROC 或者查准率-查全率曲线,将测试集中的隐藏评分合到用户-物品对的参考集中,使用推荐系统生成单一的用户-物品对的排名列表,从列表中选择最前面的推荐,在参考集上为推荐评分。以这种方法计算得到的 ROC 曲线叫作 Global ROC 曲线(GROC)[52]。在结果曲线上选取一个工作区间可能会导致不同数目的推荐给用户。

最后一类应用,其推荐过程更加具有互动性,用户可以得到越来越多的推荐。这是经典的信息检索任务,用户可以一直要求系统给出更多的文档。在这种应用中,我们为每个用户计算查准率-查全率曲线(或 ROC 曲线),然后计算所有用户结果曲线的均值。这是信息检索社区计算查准率-查全率曲线的一个常用方法,特别是在具有影响力的 TREC 比赛中[59]。这样一个曲线可以用来理解一个典型用户可能面对的查准率和查全率的折衷。

### 8.3.2.3 排名方法

在许多情况下应用通常将给用户展示的推荐设为垂直或水平列表,利用某种天然的浏览秩序。例如,在 Netflix 中,标签“你将会喜欢的电影”显示一组分类,以及每组分类中系统预测用户会喜欢的电影列表。这些列表可能很长,用户可能需要继续查看别的“页面”直到所有的列表被浏览。在这种应用中,我们感兴趣的不是之前章节中所提到的预测具体的评分或选择一组被推荐的物品等,而是根据用户偏好为物品排序。这个任务通常称为物品排名。有两种方法可以评测这种排名的准确度。我们可以尝试判断推荐给用户的物品集



的正确顺序, 评测系统距离正确的顺序。或者我们可以尝试评测系统排名的效用。我们首先介绍离线测试上的方法, 然后介绍这些方法在用户调查和在线测试上的可行性。

### 使用偏好排名

为评估相对于参考排序的排序算法(一个正确的顺序), 首先需要获取这样一个参考。在用户对物品的显性评分可获得的情况下, 我们可以将被评分的物品按照评分递减排序, 且排名可相同。例如, 在 Netflix 中, 用户排过名的电影可以以递减顺序排序, 评分同为 5 星的电影, 紧跟评分同为 4 星的电影。在我们只有使用数据的情况下, 需要构建一个参考排名, 其中用户用过的物品排在闲置物品之上。然而, 这只有在用户知道闲置物品的存在时有效, 因此我们可以推断实际上用户喜欢用过的物品而不是未用过的。因此, 例如, 在线音乐应用的日志像 Pandora<sup>⊖</sup>可以用于构建参考排名, 跟踪发现听取音乐的用户排名比跳过的用户要高。再次, 使用过的物品的排名应该相同, 闲置物品的排名也应该相同。

在以上两种参考排名中, 如果我们没有关于用户的两个物品排名有关联的信息, 那么二者排序是相同的。然而, 应用所使用的推荐系统, 如 Netflix 的“你会喜欢的电影”严格地要求物品排名不相同。在这种情况下, 在参考排名中分数相同的物品在被评估的系统中一个排在另一个之前时, 系统不应该被惩罚。归一化的基于距离的性能评测(NDPM)方法[61]适用于这种情况。如果在用户  $u$  的  $n_u$  个物品  $i$  上有参考排名  $r_{ui}$  和系统排名  $\hat{r}_{ui}$ , 那么我们可以做出如下定义:

$$C^+ = \sum_{ij} \text{sgn}(r_{ui} - r_{uj}) \text{sgn}(\hat{r}_{ui} - \hat{r}_{uj}) \quad (8.7)$$

$$C^- = \sum_{ij} \text{sgn}(r_{ui} - r_{uj}) \text{sgn}(\hat{r}_{uj} - \hat{r}_{ui}) \quad (8.8)$$

$$C^u = \sum_{ij} \text{sgn}^2(r_{ui} - r_{uj}) \quad (8.9)$$

$$C^s = \sum_{ij} \text{sgn}^2(\hat{r}_{ui} - \hat{r}_{uj}) \quad (8.10)$$

$$C^{u0} = C^u - (C^+ + C^-) \quad (8.11)$$

其中, 和的范围超过物品对数量  $\frac{1}{2}n_u(n_u - 1)$ 。  $C^u$  为参考排名所显示的排序中(例如, 排名可相同)物品对数量, 而  $C^+$  和  $C^-$  分别为系统排名所显示的正确排序和错误排序的物品对数量。  $C^{u0}$  为在参考排名中无相同排名而系统排名中有相同排名的情况下的物品对数量。于是, NDPM 可以由下式给出:

$$\text{NDPM} = \frac{C^- + 0.5C^{u0}}{C^u} \quad (8.12)$$

因此, 当系统正确的预测参考中显示每个偏好的关系时, NDPM 方法会给出一个完美的 0 分。最差的分数 1 赋给那些预测与每个参考偏好关系相矛盾的系统。只有在与参考偏好关系相反数超过一半时预测才被惩罚。预测的偏好参考未提供时(例如, 我们不知道用户真实的偏好)不受惩罚。

在某些情况下, 我们可能完全了解用户对某些物品集的真实喜好。例如, 我们可以通过为用户提供二项选择来推导用户的真实顺序。在这种情况下, 一个物品对在参考排名中排序相同意味着用户其实对两个物品不关心。因此, 一个理想的系统不应将一个物品排在另一个物品之前。排名相关性方法如 Spearman 的  $\rho$  或 Kendall 的  $\tau$  [30, 31] 可以用在这

种情况下。这些方法在实际使用中有着更高的相关性[18]。Kendall 的  $\tau$  由下式给出：

$$\tau = \frac{C^+ - C^-}{\sqrt{C^+} \sqrt{C^-}} \quad (8.13)$$

而 Spearman 的  $\rho$  由下式给出：

$$\rho = \frac{1}{n_u} \frac{\sum_i (r_{i,u} - \bar{r})(\hat{r}_{i,u} - \bar{\hat{r}})}{\sigma(r)\sigma(\hat{r})} \quad (8.14)$$

其中， $\bar{\cdot}$  和  $\sigma(\cdot)$  表示均差和标准差。注意，在排名相同时，每个物品被赋值为平均排名，例如，如果排在第二、第三位置的两个物品排名相同，那么他们的排名被赋为 2.5 [31]。

### 基于可用性的排名

在与某些“真实”排名相关性上使用参考排名对排名进行评分的同时，有其他的标准来决定物品列表的顺序。其中一个流行的方法是假设推荐列表的效用是累积的，由个人的推荐效用之和给出。每个推荐的效用为该被推荐物品的效用基于其在推荐列表中的位置打一个折扣。一个关于此效用的例子是用户注意列表中位置为  $i$  的推荐的可能性。通常假设用户是从头到尾的浏览推荐，而越往后推荐的效用的折扣越大。该折扣也可以定义为用户注意到列表中某个特定位置的推荐的概率，鉴于推荐的效用仅取决于物品推荐。在该定义中，推荐列表中的特定位置被注意到的概率假定只与位置相关而与推荐的物品无关。

在许多应用中，用户只能使用单个物品或者一个很小的物品集，或者推荐引擎没有被用作主要浏览工具。在这样的情况下，我们只能期望用户只会注意到推荐列表顶部的少数物品。我们可以使用一种随着列表上位置的下降衰减非常迅速的方法来模拟这样的应用。R-Score 度量[8]假设对于每个用户推荐值沿着排名列表以指数形式递减：

$$R_u = \sum_i \sum_j \frac{\max(r_{ui} - d, 0)}{2^{j-1}} \quad (8.15)$$

其中， $i_j$  是位置为  $j$  的物品， $r_{ui}$  是用户  $u$  对物品  $i$  的评分， $d$  是任务相关的中立（“不在乎”）评分， $\alpha$  是半衰期参数，控制着排名列表中位置值的指数衰减。在评分预测任务中， $r_{ui}$  是用户对每个物品的评分（如 4 星）， $d$  是不关心的投票（如 3 星），算法只获取分数高于“不在乎”和  $d$ （如 4 星或 5 星）的排名物品的分数。在使用预测任务中，如果用户选择物品  $i$ ， $r_{ui}$  则为 1，否则为 0， $d$  为 0。如果物品  $i$  被使用，则  $r_{ui} = -\log(\text{popularity}(i))$ ，否则为 0，这样可以从推荐中获取一定的信息量。由此生成的每个用户的得分由下式合计：

$$R = 100 \frac{\sum_u R_u}{\sum_u R_u^*} \quad (8.16)$$

其中， $R_u^*$  是用户  $u$  最可能排名的分数。

在其他应用中用户可能要阅读相当大一部分的列表。在某些类型的搜索中，例如，查找法律文件[24]，用户可能会搜索所有相关的物品，愿意阅读推荐列表的大部分。在这种情况下，位置折扣需要一个较低的衰减。归一化折扣累计收益（Normalized Discounted Cumulative Gain, NDCG）[27]是一个来自于信息检索的方法，其中位置以对数形式衰减。假设用户  $u$  被推荐物品  $i$  会得到一个“收益” $g_{ui}$ ，物品集合  $J$  的平均折扣累计收益（Discounted Cumulative Gain, DCG）可以定义为

$$\text{DCG} = \frac{1}{N} \sum_{u=1}^N \sum_{j=1}^J \frac{g_{ui}}{\max(1, \log_b j)} \quad (8.17)$$

其中对数的底是自由参数,通常在 $2\sim 10$ 。对数底为 $2$ 一般用来保证所有位置都已衰减。NDCG是DCG的归一化版本,如下式所示:

$$\text{NDCG} = \frac{\text{DCG}}{\text{DCG}^*} \quad (8.18)$$

其中是 $\text{DCG}^*$ 为理想的DCG。

我们展示这两种方法正如它们最初被展示的一样,但注意两种情况下的分子包含一个效用函数,该函数为每个物品进行赋值。我们可以使用一个对于设计的应用更合适的函数来代替原来的效用函数。一个与R-score和NDCG密切相关的方法是平均逆命中率(Average Reciprocal Hit Rank, ARHR)[14],一个没有被归一化方法,为在位置 $k$ 的成功推荐的效用赋值为 $1/k$ 。因此,ARHR衰减的比R-score慢但比NDCG快。

### 在线排名评估

在为评估推荐列表排名而设计的在线实验中,我们可以观察用户与系统的交互。在推荐列表展现在用户面前时,用户可能会从列表中选择一定量的物品。那么我们就可以假设用户至少浏览到了最后选择的位置。也就是,如果用户选择了物品1、3、10,我们就可以推断用户查看了 $1\sim 10$ 个物品。此时,我们可以做出另外一个假设,用户对1、3、10比较感兴趣,而对物品2、4、5、6、7、8、9不感兴趣。在某些情况下我们可以得到额外的信息,用户是否注意到更多的物品。例如,如果列表分散在几个页面,每页只显示20个结果,那么在上述的例子中,如果用户移到第二页,那么我们还可以假设用户浏览了结果 $11\sim 20$ ,并发现它们都是无关的。

在上面的情况中的交互结果分为三部分——感兴趣的物品(上面例子中的1、3、10),不感兴趣的物品(从 $1\sim 20$ 的其他物品),还有一些未知物品(21到列表结尾)。现在我们可以使用一个合适的参考排名指标来为原来的列表打分。有两种不同的方法可以做到。首先,参考列表可以包括顶部的感兴趣的物品,然后是未知物品和底部的不感兴趣的物品。参考列表体现了用户可能只选择一小部分的感兴趣物品,因此未知物品可能包含更多的感兴趣的物品。其次,参考列表可以将感兴趣物品置于顶部,紧接着是不感兴趣物品,而直接完全忽略未知物品。这在不合理偏好假设可能会产生消极效果时是有用的,比如,未知物品在排列中优于不感兴趣物品。不管在哪种情况,应该牢记参考排名因离线评估的情况而不同。在离线评估中,我们有一个被假设是正确的参考排名,然后衡量每个推荐偏离“正确”排名有多少。在线情况下,参考排名是显示的推荐排名中用户可能会喜欢的排名。离线情况下,我们假设有一个正确的排名,而在线情况下我们允许有多个正确的排名。

对于效用的排名,我们可以基于所选物品的效用之和对列表进行评估。因此,将效用高的感兴趣物品排名放在靠近列表开头的列表优于将感兴趣物品放在列表末尾处的列表,因为对于后者,用户一般根本不会注意那些感兴趣物品,对推荐无用。

### 8.3.3 覆盖率

正如推荐系统的预测精度,在许多情况下特别是在协同过滤系统,随着数据量的增加而增加,一些算法可能提供高质量的推荐,但只对一小部分具有大量数据的物品而言。物品覆盖率可以参考我们下面要讨论的几个不同的系统属性。

#### 8.3.3.1 物品空间覆盖率

最常见的物品覆盖率的定义为系统可以推荐的物品的比例。也经常称为目录覆盖率。

最简单的目录覆盖率的计算方法是计算所有曾经被推荐过的物品的百分比。在许多情况下该方法可以根据算法和输入的数据集进行直接计算。

一个更有用的方法是计算离线、在线或用户调查实验中被推荐给用户的物品所占的比例。有时可能要为物品加权重,如流行度或效用。因此,我们同意,无论如何也不会推荐很少被使用的物品,但忽略关注度很高的物品,这是无法容忍的。

另一个计算目录覆盖率的方法是销售多样性[16],衡量在使用特定推荐系统时用户是如何选择不同的物品的。如果对于每个物品 $i$ 用户的选择概率记为 $p(i)$ ,基尼系数由下式给出:

$$G = \frac{1}{n-1} \sum_{j=1}^n (2j-n-1)p(i_j) \quad (8.19)$$

其中, $i_1, \dots, i_n$ 为物品列表,按照概率 $p(i)$ 递增排序。当所有物品被选择的频率一样时该系数为0,如果只有一个物品被选择则该系数为1。每个物品被推荐次数的基尼系数也能用上。另外一个计算不均等分布的方法是香农熵:

$$H = - \sum_{i=1}^n p(i) \log p(i) \quad (8.20)$$

当某个物品总是被选择或者推荐时,熵为0;当 $n$ 个物品同样被选择和推荐时,熵为 $\log n$ 。

### 8.3.3.2 用户空间覆盖率

覆盖率也可以是用户或用户与可以推荐物品的系统的交互所占比例。在许多应用中推荐系统可能不会为某些用户进行推荐,比如,该用户的预测准确度置信度低。这种情况下我们比较喜欢可以为更广泛用户提供推荐的系统。很明显这样的推荐系统应该在覆盖率和准确率的折衷上进行评估。

覆盖率可以通过为获得丰富的用户画像而进行的推荐来计算。例如,在协同过滤实例中,可以计算在得到推荐之前用户必须评分的物品的数量。这种方法通常只能在离线实验中评估。

### 8.3.3.3 冷启动

另一个相关的问题是众所周知的冷启动问题——系统在遇到新物品和新用户时的性能。冷启动问题可被看作覆盖率的子问题,因为它评估的是系统在特定物品集和用户集的覆盖率。除了计算冷启动物品和用户池,为这些物品和用户计算系统的准确度也是很重要的。

针对物品冷启动,我们可以使用一个阈值来推断冷启动物品集。例如,可以推断冷启动物品是那些没有评分或没有使用证据的物品[52],或者是在系统中存在不超过某个时间的物品(如1天),又或者是在证据数量不超过预定值的物品(例如,评分不超过10)。或许更普遍的方法是通过物品在系统中存在的时间或为物品收集的数据量来判断物品是否“冷”。这样,我们可以相信系统会更多地预测合适的冷门物品,而少预测热门物品。

一个系统要更好地推荐冷门物品可能要付出降低热门物品准确度的代价。这也可能是其他的原因,例如,后面要讨论的新颖度和惊喜度。然而,在计算冷物品的系统准确度时最好评估一下与整个系统的准确度是否可以权衡。

### 8.3.4 置信度

推荐的置信度可以定义为系统在推荐和预测上的可信度[128, 22]。正如我们之前所述,协同过滤推荐的准确度随着数据量的增加而增加。同样地,被预测属性的置信度一般也随着数据量的增加而增加。

在许多情况下,用户受益于观察这些置信度分数[22]。当系统为推荐物品赋予一个低

的置信度时,用户可能会在下决定之前查看更多的物品。例如,如果系统推荐一个置信度高的电影以及一个有着相同评分但置信度低的电影,那么用户可能会立即将第一个电影加入播放列表,再进一步阅读第二个电影的剧情,在决定观看之前或许会再浏览几个电影。

最常用的置信度度量方法是计算预测值为真实的概率,或者计算落于真实值预定义部分内(如95%)的区间。例如,一个推荐系统可能以0.85的概率正确地将一个电影评分为4星,或者有95%的真实评分位于预测评分为4星的 $-1$ 到 $+\frac{1}{2}$ 区间内。计算置信度最普遍的方法是为可能的结果提供一个完整的分布[40]。

给定两个在其他相关属性上性能相似的推荐系统,如预测准确度,一般会选择可以提供有效置信度度量的系统。由此,给定两个准确度相同的推荐系统,而且使用相同方法计算置信区间,我们会倾向于选择可以更好地估计置信区间的推荐系统。

标准置信区间如上述置信区间可以在常规的离线实验中直接估计出来,与预测准确度的估计方法相同。我们可以为每个特定的置信度类型设计一个分数,用来评估置信度估计方法与真正的错误预测的差距。当算法与置信度度量方法不能达成一致时不能应用该过程,因为有些置信度评估方法比较弱因此比较容易估计。在这种情况下,更加准确地估计一个较弱的置信度度量并不意味着更好的推荐。

**例 8.1** 推荐系统 A 和 B 置信区间都落在可能的电影评分中。我们取一个范围为 95% 内的置信阈值来训练 A 和 B。对于每个训练模型,在离线数据上运行 A 和 B,隐藏一部分用户评分,让每个算法预测缺失的评分。每个算法产生一个预测评分和置信区间。我们计算  $A_+$  和  $A_-$ , 分别表示算法 A 的预测评分落在置信区间之内和区间之外的次数,对 B 做同样的计算。然后我们使用  $\frac{A_+}{A_+ + A_-} = 0.97$  和  $\frac{B_+}{B_+ + B_-} = 0.94$  计算每个算法真正的置信度。结果表明算法 A 过于保守,计算出的区间太大,而算法 B 则比较自由,计算出的区间太小。由于我们并不需要保守的区间,所以选择 B 因为它预测的区间更接近于所要求的 95% 置信度。

置信区间的另一个应用是通过预测值的置信度低于某个阈值来过滤推荐物品。在这种情况下,我们假设推荐系统并不为所有的值预测分数,因为通常情况下只显示前  $n$  个推荐。因此,我们可根据这个过滤过程设计一个实验,在两个算法过滤掉低置信度物品后比较二者的准确度。在该实验中,可以计算一个曲线用来估量过滤后物品的每个部分的预测准确度,或者不同过滤阈值的预测准确度。该评估过程并不要求两个算法都与置信度方法一致。

虽然用户调查和在线实验可以研究为用户体验加入置信度的效果,但要想知道这些类型的测试如何为置信度估量的准确度提供进一步的证据是很难的。

### 8.3.5 信任度

置信度是系统对评分的信任,而这里所说的信任度是用户对系统推荐结果的<sup>①</sup>。例如,推荐用户已经知道或喜欢的物品也许会有益于系统。这样,用户即使没有从推荐中得到有价值的东西,但他注意到了该系统提供合理的推荐,这可能会增加他对系统推荐的未知物品的信任。另一个常用的增加系统信任度的方法是解释系统提供的推荐(可参照第 15 章)。

① 不要和社交网络研究里的信任度弄混淆了,用于衡量用户有多信任另一个用户。推荐系统的一些文献中用这种信任衡量方法过滤用户,如文献[39]和第 20 章。



系统的信任度也可称为系统的信用。

如果我们不仅限于以上获取信任的方法,有一个显而易见的评估用户信任度的方法是在用户调查中询问用户该推荐系统是否合理[22, 12, 11, 47]。在线测试中,我们可以将推荐的数量与推荐在系统中的信任度相结合,假设信任度越高被使用的推荐就会越多。或者我们还可以假设系统的信任度与重复用户有关,因为信任系统的用户会返回系统执行别的任务。然而,这样的方法可能与用户满意度的其他因素相关,因此可能不准确。如何在离线实验中测量信任度还是未知的,因为信任是在用户和系统交互的过程中建立的。

### 8.3.6 新颖度

新颖的推荐是指为用户推荐他们以前没有听说过的物品[33]。在需要新颖性推荐的应用中,一个最明显简单的方法是过滤掉用户已经评分或使用过的物品。然而,在许多情况下用户并不会提供他们过去使用过的所有物品。因此,这个简单的方法并不足够将用户已经知道的物品过滤掉。

我们可以在用户调查中通过询问用户他们是否已经熟悉这个推荐物品来评测新颖度[9, 28],同时也可以通过离线实验得到一些对系统新颖性的理解。在这样一个实验中,我们可以将数据通过时间进行分割,例如,隐藏所有在某个特定时间点出现的用户评分。另外,我们可以隐藏在特定时间点之前的一些评分,来模拟用户熟悉但没有评分的物品。在进行推荐时,每个在时间片之后评分的被推荐物品将受到奖励,在时间片之前评分的被推荐物品将受到惩罚。

为了完成以上过程我们必须仔细地构建隐藏过程,这样我们可以模拟真实系统中偏好发掘过程。在某些情况下,已评分的物品并不是所有用户熟悉的物品的平均取样,应该承认并处理这种偏差。例如,如果我们认为用户会给某些特殊的物品打更高的分数,给流行物品打较低的分数,那么这个隐藏过程应该隐藏更多的流行物品。

用这种方法计算新颖度时,控制准确度是很重要的,因为无关的推荐也许对用户来说是新颖的,但是没有意义。在只考虑相关的物品的新颖度时有一个方法可以使用[63]。

**例 8.2** 我们想要在离线实验中评估一组电影推荐的新颖度。由于我们认为系统的用户在看过电影以后评分,所以将用户评分以一个连续的方式进行分割。对于每个用户资料,我们沿着电影评分的时间顺序随机选取一个分界点,隐藏在这个点之后的所有电影。

系统的用户调查表明人们并不倾向于为他们没有强烈感觉的电影打分,而偶尔也不会为他们喜欢或者不喜欢的电影打分。因此,我们将以  $1 - \frac{|r-3|}{2}$  的概率隐藏在分界点之前的电影评分,其中  $r \in \{1, 2, 3, 4, 5\}$  是电影的评分,3 是中性评分。我们要避免用隐藏评分预测这些电影,因为用户已经知道了它们。

然后,每个推荐为每个用户产生一个有 5 个推荐结果的列表,并且我们只在分界点之后的物品上计算精确度。也就是,推荐系统推荐出现在分界点之前的具有隐藏评分的电影是不得分的。在该实验中,优选精确度最高的算法。

另一个评估新颖推荐的方法是,使用上述的假设即流行的物品很少是新颖的。因此,我们可以使用这样的一种正确率度量方法来考虑新颖度,即系统对于流行商品的预测和对非流行商品的预测是不一样的[53]。Ziegler 等[64]和 Celma 和 Herrera [9]也提出了考虑概率的正确度评测方法。

最后,我们可以将推荐中新信息量与推荐物品之间的相关性一起评估。例如,当物品

评分可行时, 我们可以通过推荐物品的某个信息化方法(如给定用户资料的条件熵)乘以隐藏评分来生成新颖度分数。

### 8.3.7 惊喜度

惊喜度是成功推荐的衡量标准之一。比如, 一个用户对特定的明星演员的电影给予积极评价, 推荐该演员的新的电影就是新颖的, 因为用户可能还不知晓, 但是不足为具有惊喜度的。当然, 随机推荐也可能是惊喜的, 因此, 我们需要在惊喜度和准确度中取得平衡。

我们认为惊喜度是推荐中对用户来说比较新颖的相关信息的量。比如, 如果可以使得一系列用户喜欢的演员的电影进行推荐, 这就被认为是惊喜的。在信息检索中, 新奇性典型地被认为是在文档中找到新的信息(这最接近我们的惊喜度的定义), Zhang 等[63]建议到可以手动对一些文件标记为冗余文件。然后, 通过比较算法避免推荐冗余的文档。当一些项目具有元数据, 如内容信息时, 就可以用于这样推荐系统。

为了避免人工标记, 可以设计一种项目之间基于内容的距离测量方式(见第3章), 接着可以对成功的推荐进行评分, 这种评分主要是通过之前的协同过滤系统或从基于内容推荐的用户模型中一系列先前评分项目的距离中计算出来的[62]。因此我们对于一个成功的推荐的衡量并不仅是根据用户模型。

**例 8.3** 在一个图书推荐应用中, 我们打算从读者并不熟悉的作者中去推荐图书。因此, 我们在图书  $b$  以及一系列图书  $B$  (用户之前已经阅读过的图书) 之间设计了一个距离测度; 定义  $c_{B,w}$  为作者  $w$  在  $B$  系列图书中的图书数量。定义  $c_B = \max_w c_{B,w}$  是在  $B$  系列图书中单个作者最大图书量。定义  $d(b, B) = \frac{1 + c_B - c_{B,w}(b)}{1 + c_B}$ , 其中  $w(b)$  是图书  $b$  的作者。◀

现在开始运行一个离线实验来评估候选算法可以产生更惊喜的推荐。我们将每个受测试的用户模型分成一系列可以观察的书  $B_i^o$  和隐藏的书  $B_i^h$ , 也就是用户读的一系列书。我们使用  $B_i^o$  作为每个推荐的输入, 然后得出一列 5 个推荐。对于每个出现在用户  $i$  中的推荐列表中的隐藏的书  $b \in B_i^h$ , 推荐一个分数  $d(b, B_i^o)$ 。因此对于读者读得少的作家, 推荐会产生一些好的结果。在这个实验中, 获得更高评分的推荐被选择用来应用。

也可以认为惊喜度是“自然”预测推荐的一个偏离[44]。基于一个有较高准确率的预测引擎, 推荐结果有些太“明显”。因此, 我们会把那些预测引擎中认为不可能的成功推荐给予一个更高的惊喜度分数。

我们可以通过让一个用户去标记他们认为非期望的推荐结果, 来评估推荐的惊喜度, 然后我们可以尝试发掘是否用户顺从这些推荐结果, 这些推荐使得用户未期望的和成功的推荐被认为是惊喜度。在一个离线实验中, 我们可以认为距离测度的正确性, 评估用户模型的距离对于用户顺从的推荐的影响。利用时间来审视惊喜度的影响也是很重要的, 因为用户可能起初被未期望的推荐迷惑了, 从而尝试它们。如果接下来他们发现推荐并不是合适的, 他们可能将来不会再用该推荐了, 或者完全停止使用这种推荐引擎。

### 8.3.8 多样性

多样性一般定义成相似性的相反面。在一些情况中推荐一系列相似的项目可能对于用户没有用, 因为它可能需要更长的时间来探索范围。考虑一个度假推荐的例子[55], 系统

应该是推荐度假行程套餐。系统提出了 5 个推荐，所有都是一个地方，不同的仅是旅馆的选择或者景区的选择，所以这 5 个推荐不都是有用的。用户可以观看不同的推荐地点，然后寻求对于他合适的地方的更详细的信息。

对于测量多样性经常的方法是使用物品—物品相似性，典型是基于物品的内容，如 8.3.7 节中所述。接下来，我们可以测量基于和、平均、最小和最大物品列表的多样性，或者测量从已在列表中的物品，将每个物品加到推荐列表作为新物品的多样性的值[64, 6]。在评价中，物品—物品相似性测量与计算推荐列表的算法采用的相似性测量是不同的。比如，我们可以用代价更高的衡量标准评估，得到更精准的结果，而不是采用更适合在线计算的快速近似方法。

当多样性以耗费其他属性作为前提时，如准确性[62]，我们也可以计算曲线来评估准确率的降低和多样性的增加。

**例 8.4** 在一个图书推荐应用中，我们有兴趣向用户提供一组不同的建议，尽量以最小的影响精度为代价。我们使用例 8.3 中的  $d(b, B)$  作为距离量度。基于候选推荐，每个都含有一个可以控制推荐多样性的可调参数，我们训练每个算法在多样性参数值的范围。对于每个训练模型，我们按照以下方法计算准确的分数和多样性分数；采用一个算法生成的每个推荐列表，再计算每个物品和其他列表中距离，计算平均结果以便获得一个多样性分数。在一个图形中绘制推荐的准确性——多样性曲线，再选择主导曲线的算法。 ◀

在辅助信息搜索的推荐中(见第 18 章)，我们认为更加多样的推荐将导致更少的搜索交互[55]。我们可以在离线实验中使用该方法来测量交互序列长度作为一个多样性的代表。由于总是在线测试，短的部分可能是由于系统的其他因素，为了证实这个假设，可以用实验来证明。即在对比不同推荐之前，使用同样的预测引擎，利用不同的多样性阈值来比较。

### 8.3.9 效用

许多电子商务网站使用推荐系统是为了改善它们的营销，如提高交叉销售。在这些示例中，推荐引擎可以通过网站产生的收入来判断[54]。一般来说，我们可以定义各种推荐系统试图去最大化的效用功能。对于这样的推荐，衡量效用或者推荐的期望效用可能比衡量推荐的准确性还要重要。也可以将许多其他的属性，如多样性或“意外发现”，看作效用功能的不同类型。在本章中，我们定义效用为系统或用户从一次推荐中获取的价值。

效用可以从推荐引擎或推荐系统本身的角度来衡量。要仔细衡量用户接到的推荐的效用。首先，用户效用或偏好是很难捕捉和建模的，很多研究致力于此[7, 21, 48]。其次，汇总不同用户的效用去计算一个推荐分数是不清晰的。例如，用钱作为一个效用，因此选择一个最小用户成本的推荐是很有诱惑力的。然而，在最小回报假设中[56]，对于不同收入水平的人们而言，同样的钱没有同样的效用。因此，如每个购买的平均花费并不是一个聚集用户的合理理由。

在一个用户评价物品的应用中，使用评分作为效用测量也是可以的[8]。比如，在电影评分中，五颗星的评价被认为是优秀的电影，我们可以假设，相比于推荐四颗星，推荐一个五颗星的电影可以被认为是一个很高的效用。由于用户具有对于评分的不同解释，所以在汇总用户之前应该进行评分的标准化。

正如我们对成功的推荐分配了正效用，也可以对不成功的推荐给予效用。比如，如果某个推荐物品冒犯了用户，然后我们可以对该推荐进行负效用分配来惩罚该系统。我们可以对于每个推荐给予一定成本，可以基于列表中推荐物品的位置，也可以从物品的效用中

减去。

对于任何效用功能，推荐的标准评价是计算一个推荐的期望效用。在推荐系统尝试去预测仅仅一个单一物品时，比如，我们基于时间分割来评价系统或试图预测序列中下一个物品时，一个合适的推荐值应该是物品的效用。当推荐系统预测  $N$  个物品时，我们可以使用列表中正确的推荐效用总和。当对于不好的推荐使用一个负效用时，就是对所有的推荐的总和，无论是成功还是失败的推荐。我们可以将效用整合到排名测量中，正如 8.3.2.3 节中所讨论的。最后，我们可以基于最优的推荐列表使用最大可能效用来使得结果评分标准化。

说到推荐效用，在用户调查和在线实验情况下评价效用是很容易的。如果我们最大的效用是网站的收入，衡量各种推荐系统的用户之间收入变化是很容易的。如果优化在线评价用户效用变得困难，那是因为用户发现对于结果的效用分配是很难的。然而，在许多情况下，用户可以说是否喜欢一个结果，因此，为了排名候选的方法，我们可以尝试引出用户偏好[26]。

### 8.3.10 风险

在一些情况下，一个推荐可能会被认为是有潜在风险的。例如，当推荐股票时，用户希望是规避风险的，希望更低期待增长的股票，但是最好也是低风险的。另一方面，用户可能是寻求风险的，偏爱那些有潜在高回报的股票。在这种情况下，我们可能希望去评价不仅从推荐中产生的价值，而且还希望能有最小风险的。

标准的评价风险敏感度的系统，不仅考虑期望效用，还有效用方差，例如，我们可以使用一个参数  $q$  比较两个系统的  $E[X] + q \cdot \text{Var}(X)$ 。当  $q$  是负的时，系统就会推荐规避风险的。

### 8.3.11 健壮性

稳健性(见第 25 章)是指在出现虚假信息的情况下推荐的稳定性[45]，尤其是被故意插入为了影响推荐的虚假信息。随着越来越多的人依赖推荐系统来引导他们，通过影响系统来更改某一物品的评级对利害关系人可能是有利可图的。例如，酒店业主希望能够提高他们酒店的评级。这可以通过注入积极评价本酒店或者消极评价竞争对手的虚假的用户信息来实现。

这种影响推荐的企图通常被称为攻击[43, 35]。当一个恶意用户有意地查询数据集或者注入虚假信息以了解一些用户的私人信息时，协同攻击就会发生。在评价这样的系统时，由于系统的灵敏度因协议的不同而不同，所以提供一个完整的攻击协议说明书是十分重要的。

通常，创建一个能免除任何攻击的系统是不切实际的。在大多数情况下，能够注入无限量信息的攻击者可以以任意的方式操纵一项推荐。因此，估计影响推荐的成本是十分有用的，尤其是推荐是由大量的注入信息来决定的。尽管理想的情况是从理论上分析修改评级的成本，但是这并不总是可行的。在这些情况下，我们可以通过向系统的数据集中导入虚假信息来模拟一组攻击，实证测量一次成功攻击的平均成本[36, 10]。

相对于在这里讨论其他的评价标准，对一个真实系统进行攻击作为一次在线实验是很难想象的。然而，通过分析从在线系统收集的真实数据来识别针对系统执行的实际攻击，

可能是有成效的。

另一种类型的稳健性是指在极端条件下的稳定性,如大量请求发生时。虽然讨论较少,但是这种稳健性对系统管理员是非常重要的,他们必须避免系统故障。在许多情况下,系统稳健性和基础设施是相关的,如数据库软件或者硬件配置;与可扩展性也是相关的(见 8.3.14 节)。

### 8.3.12 隐私

在一个协同过滤系统中,一个用户愿意去说出他的物品偏好以期望得到好的推荐。然后,对于许多用户来说,保持隐私也是很重要的,即没有第三方可以使用推荐系统去了解一些特殊用户的某些偏好。

例如,考虑一个用户非常喜欢他买的一本书《离婚的组织 and 计划》,他的配偶可能会被推荐“买这本书也会购买这个本书”,这样就暴露了一些个人的隐私信息。

一般来讲,一个推荐系统泄露私人信息,甚至是个人隐私,都是不适当的。出于这些原因,分析隐私一般趋向于聚焦在不好的情景中,在用户隐私信息可能泄露的情况下说明了一些事实。其他一些研究者[17]通过评估那些隐私信息被泄露的用户的比例来比较算法。在这样的研究中,假设完全的隐私是不现实的,因此我们必须减少侵犯隐私。

其他可代替的就是定义不同的隐私程度,如  $k$  识别[17]以及对于不同程度的隐私侵犯去比较他们的算法敏感度。

隐私的保护需要以推荐的准确率作为代价,因此,必须在这之间寻找一个平衡。或许最重要的实验是,当隐私修正被增加到算法中时,评估准确性(或其他平衡属性)可以需要或不需要修正信息[42]。

### 8.3.13 适应性

现实的推荐系统可以在那些物品集有很大变化的环境中或兴趣趋势经常改变的环境中运行。或许最明显的例子就是推荐一个新物品或在线报纸的相关故事[19]。在这种情景中,故事可能是仅在短时间内有兴趣,之后就会变得过时。当未预料的新事件发生时,如台风,人们会对这类事感兴趣,也会对一些相关的旧事物感兴趣,如过去的关于台风的新闻。尽管这类问题类似于冷启动问题,但是不完全一样,因为这些过去的没有作为感兴趣的新闻如今变得让人感兴趣了。

这种类型的适应性可以通过分析推荐之前的信息数量来离线评价。如果我们用连续方式对推荐过程建模,我们可以记录推荐一个故事前的所需要的算法证据。一个算法可以快速适应推荐的物品,只要他们使得有兴趣就可以,但是需要牺牲推荐的准确性。我们可以通过评估准确率和趋势改变的速度之间的平衡来比较两种算法。

另外一种适应性类型是适应用户的个人偏好进行评分[37],或者是在用户模型中改变[34]。例如,当用户对一个物品进行评分时,他们也期望一系列的推荐可以改变。如果推荐的结果保持不变,用户会认为他们的评分的努力是一种浪费,以至于将不会继续评分。随着这种趋势评估的改变,我们可以对用户模型增加更多的信息后(如新的评分),在离线实验中再次评估这种推荐列表的改变。我们可以通过测量推荐列表在增加新信息之前和之后的不同来评估算法。8.3.3 节的基尼系数和香农熵的例子可以用来测量当用户模型变化时,对于用户的推荐的变化。



### 8.3.14 可扩展性

当推荐系统可以设计成帮助用户在大的物品集合中进行导航,设计这样的系统的目标的人也期望扩展到数据集中。因此,通常情况下,算法用来替换其他的属性,如,准确率或覆盖率。甚至对于庞大的数据集也可以提供快速的结果(如文献[12])。

随着数据集的增长,许多算法要么变得缓慢,要么需要额外的资源,如计算能力或内存。一个突出的方法是就所需要的空间和时间来计算一个算法的复杂度[29, 4]。然而,在许多情况下,两个算法的复杂度要么是一样的,要么是可以用过改变一些参数降低复杂度,如模型的复杂度或样本大小。因此,为了了解系统的可扩展性,记录下系统在大数据集下的系统消耗是很有用的。

可扩展性的测量主要是通过增长的大数据集的实验来获得的,当任务进行扩展时(如文献[19]),显示出速度和资源的消耗行为。对于测量可扩展性的支配协调也是很重要的,例如,如果算法的准确性低于在一个在相对的小数据集上的候选算法,就要展示出准确率的不同。这种测量可以在如特定人物或某个探索方向上的推荐系统的潜力表现中提供重要的有价值的信息。

当推荐系统期望在许多情况下可以提供快速的在线推荐时,测量这种系统提供推荐的速度是很重要的[23, 50]。一个测量标准是系统的吞吐量,如系统每秒提供的推荐的数量。我们也可以测量延迟时间,即在线推荐所需要的时间。

## 8.4 总结

本章讨论为了选择一个候选对象的最佳推荐算法,以及推荐算法是如何被评价的。这对于在实验中尝试找到更好的算法以及在为一个应用选择已存在的算法时是很重要的步骤。因此,许多评价机制在过去都已经被应用于算法选择了。

我们同时也考虑了设计在线和离线以及用户实验时的不同想法。本章大致描述了一些重要的测量方法,以及必须考虑除了提供的分数度量,也应该有在设计实验时选择的算法。

我们特定了一系列属性,这些有时对推荐系统是很重要的,对于每个这样的属性,我们都提出相关实验基于该属性对各种推荐系统进行排名。对于少用的那些属性,我们将它们限制到可以应用到这种属性的各种表现的通用描述上。能够实际实现的特定程序可以基于我们的通用准则为特定的属性表现而进行开发。

## 参考文献

1. Bamber, D.: The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *Journal of Mathematical Psychology* **12**, 387–415 (1975)
2. benjamini: Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Statist. Soc. B* **57**(1), 289–300 (1995)
3. Bonhard, P., Harries, C., McCarthy, J., Sasse, M.A.: Accounting for taste: using profile similarity to improve recommender systems. In: *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp. 1057–1066. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1124772.1124930>
4. Boutilier, C., Zemel, R.S.: Online queries for collaborative filtering. In: *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics* (2002)
5. Box, G.E.P., Hunter, W.G., Hunter, J.S.: *Statistics for Experimenters*. Wiley, New York (1978)

6. Bradley, K., Smyth, B.: Improving recommendation diversity. In: Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, pp. 85–94 (2001)
7. Brazunas, D., Boutilier, C.: Local utility elicitation in GAI models. In: Proceedings of the Twenty-first Conference on Uncertainty in Artificial Intelligence, pp. 42–49. Edinburgh (2005)
8. Breese, J.S., Heckerman, D., Kadie, C.M.: Empirical analysis of predictive algorithms for collaborative filtering. In: UAI, pp. 43–52 (1998)
9. Celma, O., Herrera, P.: A new approach to evaluating novel recommendations. In: RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems, pp. 179–186. ACM, New York, NY, USA (2008). DOI <http://doi.acm.org/10.1145/1454008.1454038>
10. Chirita, P.A., Nejdl, W., Zamfir, C.: Preventing shilling attacks in online recommender systems. In: WIDM '05: Proceedings of the 7th annual ACM international workshop on Web information and data management, pp. 67–74. ACM, New York, NY, USA (2005). DOI <http://doi.acm.org/10.1145/1097047.1097061>
11. Cramer, H., Evers, V., Ramlal, S., Someren, M., Rutledge, L., Stash, N., Aroyo, L., Wielinga, B.: The effects of transparency on trust in and acceptance of a content-based art recommender. *User Modeling and User-Adapted Interaction* **18**(5), 455–496 (2008). DOI <http://dx.doi.org/10.1007/s11257-008-9051-3>
12. Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization: scalable online collaborative filtering. In: WWW '07: Proceedings of the 16th international conference on World Wide Web, pp. 271–280. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1242572.1242610>
13. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **7**, 1–30 (2006)
14. Deshpande, M., Karypis, G.: Item-based top-N recommendation algorithms. *ACM Transactions on Information Systems* **22**(1), 143–177 (2004)
15. Fischer, G.: User modeling in human-computer interaction. *User Model. User-Adapt. Interact.* **11**(1-2), 65–86 (2001)
16. Fleder, D.M., Hosanagar, K.: Recommender systems and their impact on sales diversity. In: EC '07: Proceedings of the 8th ACM conference on Electronic commerce, pp. 192–199. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1250910.1250939>
17. Frankowski, D., Cosley, D., Sen, S., Terveen, L., Riedl, J.: You are what you say: privacy risks of public mentions. In: SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 565–572. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1148170.1148267>
18. Fredricks, G.A., Nelsen, R.B.: On the relationship between spearman's rho and kendall's tau for pairs of continuous random variables. *Journal of Statistical Planning and Inference* **137**(7), 2143–2150 (2007)
19. George, T.: A scalable collaborative filtering framework based on co-clustering. In: Fifth IEEE International Conference on Data Mining, pp. 625–628 (2005)
20. Greenwald, A.G.: Within-subjects designs: To use or not to use? *Psychological Bulletin* **83**, 216–229 (1976)
21. Haddawy, P., Ha, V., Restificar, A., Geisler, B., Miyamoto, J.: Preference elicitation via theory refinement. *Journal of Machine Learning Research* **4**, 2003 (2002)
22. Herlocker, J.L., Konstan, J.A., Riedl, J.T.: Explaining collaborative filtering recommendations. In: CSCW '00: Proceedings of the 2000 ACM conference on Computer supported cooperative work, pp. 241–250. ACM, New York, NY, USA (2000). DOI <http://doi.acm.org/10.1145/358916.358995>
23. Herlocker, J.L., Konstan, J.A., Riedl, J.T.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Inf. Retr.* **5**(4), 287–310 (2002). DOI <http://dx.doi.org/10.1023/A:1020443909834>
24. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* **22**(1), 5–53 (2004). DOI <http://doi.acm.org/10.1145/963770.963772>
25. Hijikata, Y., Shimizu, T., Nishida, S.: Discovery-oriented collaborative filtering for improving user satisfaction. In: IUI '09: Proceedings of the 13th international conference on Intelligent user interfaces, pp. 67–76. ACM, New York, NY, USA (2009). DOI <http://doi.acm.org/10.1145/1502650.1502663>
26. Hu, R., Pu, P.: A comparative user study on rating vs. personality quiz based preference elicitation methods. In: IUI '09: Proceedings of the 13th international conference on Intelligent user interfaces, pp. 367–372. ACM, New York, NY, USA (2009). DOI <http://doi.acm.org/10.1145/1502650.1502663>

- 1145/1502650.1502702
27. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.* **20**(4), 422–446 (2002). DOI <http://doi.acm.org/10.1145/582415.582418>
  28. Jones, N., Pu, P.: User technology adoption issues in recommender systems. In: *Networking and Electronic Conference* (2007)
  29. Karypis, G.: Evaluation of item-based top-n recommendation algorithms. In: *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pp. 247–254. ACM, New York, NY, USA (2001). DOI <http://doi.acm.org/10.1145/502585.502627>
  30. Kendall, M.G.: A new measure of rank correlation. *Biometrika* **30**(1–2), 81–93 (1938)
  31. Kendall, M.G.: The treatment of ties in ranking problems. *Biometrika* **33**(3), 239–251 (1945)
  32. Kohavi, R., Longbotham, R., Sommerfield, D., Henne, R.M.: Controlled experiments on the web: survey and practical guide. *Data Min. Knowl. Discov.* **18**(1), 140–181 (2009). DOI <http://dx.doi.org/10.1007/s10618-008-0114-1>
  33. Konstan, J.A., McNee, S.M., Ziegler, C.N., Torres, R., Kapoor, N., Riedl, J.: Lessons on applying automated recommender systems to information-seeking tasks. In: *AAAI* (2006)
  34. Koychev, I., Schwab, I.: Adaptation to drifting user's interests. In: *Proceedings of ECML2000 Workshop: Machine Learning in New Information Age*, pp. 39–46 (2000)
  35. Lam, S.K., Frankowski, D., Riedl, J.: Do you trust your recommendations? an exploration of security and privacy issues in recommender systems. In: *Proceedings of the 2006 International Conference on Emerging Trends in Information and Communication Security (ETRICS)* (2006)
  36. Lam, S.K., Riedl, J.: Shilling recommender systems for fun and profit. In: *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pp. 393–402. ACM, New York, NY, USA (2004). DOI <http://doi.acm.org/10.1145/988672.988726>
  37. Mahmood, T., Ricci, F.: Learning and adaptivity in interactive recommender systems. In: *ICEC '07: Proceedings of the ninth international conference on Electronic commerce*, pp. 75–84. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1282100.1282114>
  38. Marlin, B.M., Zemel, R.S., Roweis, S., Slaney, M.: Collaborative filtering and the missing at random assumption. In: *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence* (2007)
  39. Massa, P., Bhattacharjee, B.: Using trust in recommender systems: An experimental analysis. In: *Proceedings of iTrust2004 International Conference*, pp. 221–235 (2004)
  40. McLaughlin, M.R., Herlocker, J.L.: A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In: *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 329–336. ACM, New York, NY, USA (2004). DOI <http://doi.acm.org/10.1145/1008992.1009050>
  41. McNee, S.M., Riedl, J., Konstan, J.A.: Making recommendations better: an analytic model for human-recommender interaction. In: *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pp. 1103–1108. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1125451.1125660>
  42. McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the netflix prize contenders. In: *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 627–636. ACM, New York, NY, USA (2009). DOI <http://doi.acm.org/10.1145/1557019.1557090>
  43. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.* **7**(4), 23 (2007). DOI <http://doi.acm.org/10.1145/1278366.1278372>
  44. Murakami, T., Mori, K., Orihara, R.: Metrics for evaluating the serendipity of recommendation lists. *New Frontiers in Artificial Intelligence* **4914**, 40–46 (2008)
  45. O'Mahony, M., Hurley, N., Kushmerick, N., Silvestre, G.: Collaborative recommendation: A robustness analysis. *ACM Trans. Internet Technol.* **4**(4), 344–377 (2004). DOI <http://doi.acm.org/10.1145/1031114.1031116>
  46. Pfleeger, S.L., Kitchenham, B.A.: Principles of survey research. *SIGSOFT Softw. Eng. Notes* **26**(6), 16–18 (2001). DOI <http://doi.acm.org/10.1145/505532.505535>
  47. Pu, P., Chen, L.: Trust building with explanation interfaces. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 93–100. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1111449.1111475>
  48. Queiroz, S.: Adaptive preference elicitation for top-k recommendation tasks using gain-networks. In: *AIAP'07: Proceedings of the 25th conference on Proceedings of the 25th*

- IASTED International Multi-Conference, pp. 579–584. ACTA Press, Anaheim, CA, USA (2007)
49. Salzberg, S.L.: On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Min. Knowl. Discov.* **1**(3), 317–328 (1997). DOI <http://dx.doi.org/10.1023/A:1009752403260>
  50. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pp. 285–295. ACM, New York, NY, USA (2001). DOI <http://doi.acm.org/10.1145/371920.372071>
  51. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for e-commerce. In: *EC '00: Proceedings of the 2nd ACM conference on Electronic commerce*, pp. 158–167. ACM, New York, NY, USA (2000). DOI <http://doi.acm.org/10.1145/352871.352887>
  52. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 253–260. ACM, New York, NY, USA (2002). DOI <http://doi.acm.org/10.1145/564376.564421>
  53. Shani, G., Chickering, D.M., Meek, C.: Mining recommendations from the web. In: *RecSys '08: Proceedings of the 2008 ACM Conference on Recommender Systems*, pp. 35–42 (2008)
  54. Shani, G., Heckerman, D., Brafman, R.I.: An mdp-based recommender system. *Journal of Machine Learning Research* **6**, 1265–1295 (2005)
  55. Smyth, B., McClave, P.: Similarity vs. diversity. In: *ICCBR*, pp. 347–361 (2001)
  56. Spillman, W., Lang, E.: *The Law of Diminishing Returns*. World Book Company (1924)
  57. Swearingen, K., Sinha, R.: Beyond algorithms: An HCI perspective on recommender systems. In: *ACM SIGIR 2001 Workshop on Recommender Systems* (2001)
  58. Van Rijsbergen, C.J.: *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA (1979). URL <http://portal.acm.org/citation.cfm?id=539927>
  59. Voorhees, E.M.: Overview of trec 2002. In: *Proceedings of the 11th Text Retrieval Conference (TREC 2002)*, NIST Special Publication 500-251, pp. 1–15 (2002)
  60. Voorhees, E.M.: The philosophy of information retrieval evaluation. In: *CLEF '01: Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, pp. 355–370. Springer-Verlag, London, UK (2002)
  61. Yao, Y.Y.: Measuring retrieval effectiveness based on user preference of documents. *J. Amer. Soc. Inf. Sys* **46**(2), 133–145 (1995)
  62. Zhang, M., Hurley, N.: Avoiding monotony: improving the diversity of recommendation lists. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 123–130. ACM, New York, NY, USA (2008). DOI <http://doi.acm.org/10.1145/1454008.1454030>
  63. Zhang, Y., Callan, J., Minka, T.: Novelty and redundancy detection in adaptive filtering. In: *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 81–88. ACM, New York, NY, USA (2002). DOI <http://doi.acm.org/10.1145/564376.564393>
  64. Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pp. 22–32. ACM, New York, NY, USA (2005). DOI <http://doi.acm.org/10.1145/1060745.1060754>

## IPTV 服务提供商推荐系统： 一个大规模真实产品环境的应用

Riccardo Bambini、Paolo Cremonesi 和 Roberto Turrin

**摘要** 在本章将介绍 Fastweb。Fastweb 是欧洲最大的网络电视(IPTV)服务提供商之一，在电视网络生产环境中集成了推荐系统。该推荐系统同时采用了协同过滤和基于内容的技术，为了满足 IPTV 架构的需要(比如，受限的屏幕设定、简化的导航能力以及严格的时间限制)做了适当的调整。这些算法广泛采用线上线下测试分析，数据显示推荐系统带来了很好的效果：超过 30% 的推荐转为了购买行为，销售量提升约 15%。

### 9.1 简介

IP 电视(IPTV)通过宽带因特网[23, 17]以数码方式传播多媒体内容(如电影、新闻报道、纪录片等)。IPTV 服务包括预定电视节目以及视频点播内容[30]。在本章其余部分我们将预定电视节目和视频点播内容统称为物品。

本章介绍 Fastweb 与 Neptuny 的 ContentWise 推荐系统的集成方案。Fastweb 成立于 2001 年，是全球第一家推出完全基于 IP 的宽带电视服务公司。Fastweb 节目单包含数千套多媒体节目内容，服务着几十万 IPTV 客户。从 2007 年起 Fastweb 成为瑞士电信(Swisscom)集团的一部分。ContentWise 推荐算法是与米兰理工学院(Politecnico di Milano.)计算机系合作共同开发的。

与传统的电视不同，IPTV 为所有可访问的内容[13]提供可交互的导航服务，同时，特别是 IPTV 能够收集隐式的使用数据以及显式的用户偏好数据，从而为用户提供个性化的导航服务。用户通过一种特殊的电子设备——机顶盒(STB)——与 IPTV 系统进行交互。机顶盒所具有若干特性也限制了用户的交互，主要有：1)用户通过遥控器来控制机顶盒，而遥控器能执行的操作很有限；2)电视中显示的用户界面被设计成可以在比 PC 和用户间距离更远的距离处观看；3)这个系统主要处理多媒体内容，而这些内容的导航由于频道切换时间导致响应不够快速。

与传统的电商领域(如 Amazon、Netflix、iTunes、IMDB、Last.fm)所采用的推荐系统不同，IPTV 推荐系统必须满足特殊的需求：

- 受限于屏幕大小以及导航能力较弱，推荐物品列表必须较少；
- 必须在严格的时间(如毫秒级)限制内生成推荐物品，因为电视用户习惯于快速响应的系统；
- 系统需要能够随着用户量的增长以及节目单内容的增加进行合理的扩展；

---

Riccardo Bambini, Fastweb, via Francesco Caracciolo 51, Milano, Italy e-mail: riccardo.bambini@fastweb.it  
Paolo Cremonesi, Roberto Turrin, Politecnico di Milano, p. zza Leonardo da Vinci 32, Milano, Italy Neptuny, via  
Durando 10, Milano, Italy e-mail: paolo.cremonesi@polimi.it e-mail: roberto.turrin@polimi.it

翻译：刘良良 审核：邓展成，张杰，严强



- 由于存在直播频道, 所以部分目录应是高度动态变化的。

Fastweb 中部署的推荐系统采用两种协同过滤算法(基于物品间相似度以及降维技术)和一种基于内容的算法(基于隐语义分析)产生推荐结果。推荐系统根据上下文选择合适的算法。例如, 如果用户正在读电影简介, 查找自己所喜欢演员的电影, 那么就采用基于内容的算法。为了满足严格的实时要求, 推荐系统以及相应的算法采用基于模型的方式并且在逻辑上被划分为两个异步阶段, 批处理阶段和实时阶段。

整个推荐系统架构的输入数据主要由两部分组成: 1) 物品—内容矩阵; 2) 用户—评分矩阵。物品—内容矩阵(ICM)描述每个物品的主要属性(元数据), 如电影标题、演员列表以及电影类型。用户—评分矩阵(URM)搜集用户关于物品的评分(如偏好)。用户评分主要是隐式的, 例如, 系统可以判断出一个用户是否看了某个节目而不知道用户对该节目的显式评价。

在推荐系统部署到生产环境中前, 已经通过  $K$  折交叉验证进行大量的性能分析。结果显示对于基于内容的算法的召回率为 2.5%, 而协同过滤算法的召回率能够超过 20%。

该推荐系统在 2008 年 10 月被发布到生产环境中并能够为其中一个 Fastweb 视频点播内容类别提供服务。系统平均每天提供 30 000 个推荐并且在高峰时期每分钟可以提供 120 个推荐结果。线上分析显示几乎 20% 的推荐能够转化为用户购买行为。提升因子(如 VOD 销售量的增加)估计为 15%。

剩余章节安排如下: 9.2 节描述一个 IPTV 服务的典型架构。9.3 节描述推荐系统的架构。9.4 节描述实现的推荐算法。9.5 节描述 Fastweb IPTV 架构中实现的推荐服务。9.6 节评价推荐质量。最后, 9.7 节总结本章内容。

## 9.2 IPTV 架构

IPTV, 又称为互联网协议电视, 是建立在基于 IP 的专用宽带网络之上的视频服务, 能够提供高质量的传统电视频道节目和音视频点播内容。从终端用户的角度看, IPTV 看上去像一个标准的电视服务。在部署 IPTV 服务时, IPTV 提供商就包括从有线和卫星电视网络传输公司到大型电话公司和专有网络运营商。IPTV 有以下独特的特点[13]:

- **支持交互的电视:** 与传统单向传输的电视不同, IPTV 的双向传输能力允许用户与系统进行交互;
- **时间位移:** IPTV 通过个人视频录像(PVR)功能——一种记录并存储 IPTV 内容以便以后回放的机制——支持节目时间导航(如快进、暂停、后退)。
- **个性化:** IPTV 通过让用户自己决定他们想看什么以及什么时候看, 使终端用户个性化其电视观看体验。

图 9.1 表示一种通用的 IPTV 系统架构, 该架构支持直播电视频道(也叫作线性频道)和视频点播技术(VOD)。广播电视服务表现为用户同步接收免费的或付费观看的传统电视频道。视频点播服务表现为观看由服务提供商根据需求定制提供的多媒体内容。

IPTV 数据中心(又叫作头端)从不同的数据源, 包括地面载体、卫星载体以及传送电缆接收直播频道。一旦接收到数据, 许多不同的硬件组件, 从编码器到视频服务器, 会被用于在 IP 网络传输视频内容做准备。点播的内容会存储在快速存储盒(如固态硬盘)中。

机顶盒(STB)是一个连接网络和家庭电视的电子设备, 它负责处理接收到的信息流并将结果视频显示在电视上。用户通过手持式遥控器与机顶盒交互。遥控器使用户能够操作机顶盒的附加功能, 如电子节目向导(EPG), 而该向导是一个在接下来的一段时间(一般为 36 小时或更多)可访问的频道或节目列表。

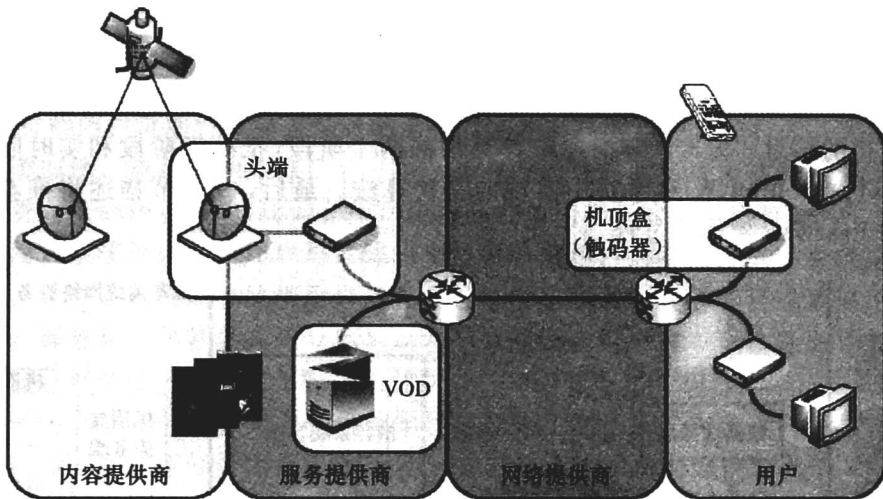


图 9.1 IPTV 系统架构

### 9.2.1 IPTV 搜索问题

为了从大量的 IPTV 频道和内容中获益，用户需要快速而且简单地查找他们实际感兴趣的東西，并且让他们能够在起居室的沙发上休息时也能不费力地做到这些，而在起居室里没有键盘、鼠标以及屏幕等桌面网页浏览时典型的设备。然而，对 IPTV 用户来说，搜索直播频道或点播视频内容是一个挑战性的问题[11]。

当观看直播电视时，用户浏览所有可观看的频道直到找到喜欢的东西。频道选择(跳过广告节目)包含两个步骤：1)浏览频道内容来决定是否继续或停下观看该频道；2)切换不同的频道重复浏览直到找到想看的频道。随着现代 IPTV 系统提供的频道数量的增加，快速找到想看的频道变得越来越困难。而且，与传统电视相比，IPTV 由于受一些技术的限制[19]，在其频道切换时需要一段时间，因此无法及时响应。

当搜索点播内容时，IPTV 用户通常要通过一个复杂的、预定义的并且时常深层嵌套的菜单结构进行导航，或者使用屏幕键盘或遥控器键盘上三击输入框输入标题。这些交互界面比较复杂并且不容易随着可获得的内容范围的扩大而扩展。此外，对于传统的个人电脑屏幕来讲，电视屏幕分辨率是有限的，使得传统图形用户界面难以使用。

这区别于传统的基于网络的领域(如电子商务站点)，这些领域的内容是文本的，适合信息分类以及关键字检索，并且输入设备(键盘和鼠标)允许在屏幕中输入任意的主题同时容易输入文本。

推荐系统与 IPTV 基础架构的整合为用户提供了一种新的更有效的浏览用户感兴趣的节目和电影的方式，从而提升用户体验。但是，这种集成必须解决以下问题：

- **用户识别**：STB 通常被家庭所有成员使用，IPTV 推荐系统无法识别到底是谁在观看某个节目；
- **实时要求**：IPTV 推荐系统必须在非常严格的实时条件限制(几毫秒)下生成推荐列表以避免减慢用户的导航过程，而该过程受到较长的频道切换时间的影响。
- **内容元数据质量**：与基于网络的领域不同，基于内容的 IPTV 推荐算法利用低质量的元数据。这种情况在直播频道中尤为明显，在直播频道中每天都会以非常高的频率增加新的内容，唯一能够描述这些节目的元数据可以在电子节目向导中找到。

### 9.3 推荐系统架构

图 9.2 表示 Fastweb 推荐系统架构, 这些组件在以下章节中讨论。9.3.1 节描述推荐系统可以获得的信息。9.3.2 节描述推荐算法的两个阶段(批处理阶段和实时阶段)架构。9.4 节详细讨论 ContentWise 系统中实现的三种算法。最后, 9.5 节描述将推荐系统整合入已存在的 Fastweb 架构的过程。

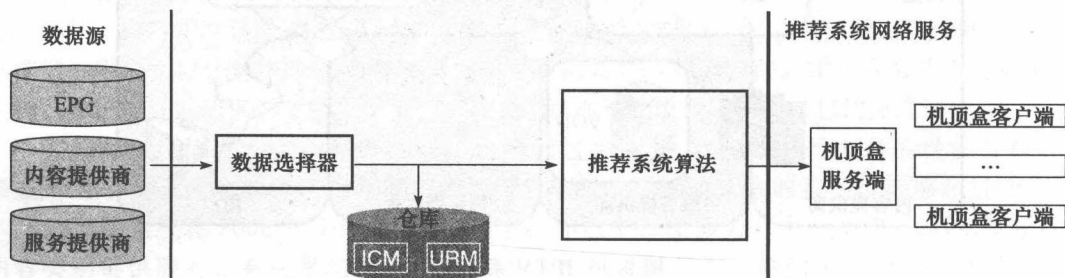


图 9.2 ContentWise 推荐系统架构

#### 9.3.1 数据搜集

负责数据预处理以及生成推荐系统输入的逻辑模块被称为**数据收集器**。数据收集器从不同的数据源, 比如, 有关直播节目信息的电子节目向导、视频点播内容信息提供商以及用户信息服务提供商等处汇集数据。

Fastweb 推荐系统不依赖于用户的个人信息(如年龄、性别、职业)。推荐过程基于用户以往的行为(他们看了什么)以及他们所表达的显式偏好(如类别偏好)。如果用户没有明确的偏好, 系统可以根据用户过去的活动推断出其偏好。

9.2 节提出了一个重要问题: 用户通过机顶盒与 IPTV 系统交互, 但是一般而言我们无法确定到底是谁站在电视前。结果, STB 搜集了一组用户的行为与偏好(比如, 家庭的部分成员)。这是一个必须考虑的问题, 因为我们只能为每个 STB 进行推荐。为了简化标记, 下面我们将用户和 STB 看作相同的实体。用户模糊问题可以采用将信息根据时间片分段的方法部分解决。例如, 我们可以粗略地设想以下场景: 家庭主妇习惯早上看电视, 孩子喜欢在下午, 晚上一般是全家一块, 而只有成人会在深夜里看电视。通过这种简单的时间段区分方式我们能够在同一个 STB 中区分不同的潜在用户。

正常情况下, 得到的信息被分成 2 个主要的矩阵: 物品—内容矩阵(ICM)和用户—评分矩阵(URM), 实际上被存储在关系数据库中。

前者描述了每个物品的主要特征(元数据)。下面我们会将物品—内容矩阵记作  $W$ , 其中元素  $w_{ci}$  表示特征(元数据)  $c$  和物品  $i$  的相关性。ICM 是通过分析内容提供商(如 EPG)所提供的信息生成的。这些信息主要关注电影名称、演员、导演、类型和剧情等。注意在真实的环境中, 特别是由于每天增加新内容的比例, 我们需要面对不确定的信息。ICM 提供的信息在经过 PoS(词性)标记, 去除停用词以及隐语义分析[32]等技术处理后用于基于内容的推荐。此外, ICM 可以用于对物品进行一些处理(如父母控制)。

URM 表示用户对物品的打分(偏好)。下面我们将该矩阵记作  $R$ , 其中元素  $r_{pi}$  代表用户  $p$  对物品  $i$  的打分。这些偏好信息构成了协同过滤算法的基本信息。用户打分可以是显

式的或隐式的，分别根据用户是否显式地表达其偏好或系统隐式地搜集进行判断。

虽然显式评分会有偏差[4]，比如，用户主观性、物品流行度以及全局评分趋势的影响，它比较确定地表示用户的观点。第一种偏见依赖评分范围的任意解释。比如，打分范围为1~5，有些用户可能打3分表示对物品感兴趣而另一些用户可能打3分表示不太喜欢这些物品。类似地，流行的物品容易被打高分而不太流行的物品容易被打低分。最后，显式评分容易受大众的观点影响(比如，用户更容易给他们都喜欢的电影打分)。

另一方面，系统可以根据用户与系统的交互推断出用户的隐式评分，尽管这些评分未必是用户的。比如，系统能够监测用户是否观看了指定频道的直播节目或者该用户是否不间断地看了一部电影。尽管显式评分比隐式评分更能表达用户对一个物品的喜好，然而从用户角度来看，显式评分的搜集比较令人讨厌。

当前部署的 Fastweb 推荐系统仅仅搜集隐式评分，但是系统在隐式评分与显式评分同时存在时也能够工作。评分范围是1~5，小于3的分值表示负面评价而大于3的分值表示正面评价。除了显式评分，系统通过监测用户行为推断出的评分被认为是正的(大于3)。实际上，不论用户是否开始看某个节目，该节目一定有一些特征信息(如演员或类型)会吸引该用户。在一些著名的显式数据集，比如，Netflix 和 MovieLens 中平均打分大于3的事实证实了这个假设。系统对 IPTV 直播节目和视频点播进行不同的处理：

**IPTV 节目：**评分与用户播放时间，比如，用户已经播放部分的百分比，呈正比例关系([18, 35])。假设  $L$  是节目的总时间， $t$  是用户播放的时间。播放时间小于5分钟的被忽略不计。如果用户看完了整个节目则分值为5，如果观看5分钟则分值为3，否则分值为3~5，计算公式如下：

$$\hat{r} = 3 + 2 \frac{t-5}{L-5}, 5 \leq t \leq L \quad (9.1)$$

$t$  和  $L$  的单位是分钟。

在该项目的开始阶段，主要的目标不是定义准确的隐式评分机制而是过滤掉干扰信息(比如，跳过广告节目)。

**视频点播电影：**当看一部电影时，用户被明确要求购买和付费。因此，独立于用户播放时间，每当用户请求观看一部电影时，系统为该电影赋予4分的隐式评分。

正如前面提到的，如果 Fastweb 也开始搜集显式评分，他们在 URM 中可以和隐式评分自然地共存。

存储在 URM 中的分值，在用于推荐算法之前会减去一个常数2.5进行标准化。这使得算法能够区分正分值和负分值，因为分值大于等于2.5(如3, 4, 5)的还是正值，而小于2.5(如1, 2)变成了负值。在剩余章节我们假设为推荐算法输入的是标准化的 URM。

最后，用户可以为他们想看的内容表达**显式的偏好**。比如，通过图形界面用户可以设置他喜欢的演员。9.4.2节讨论的基于内容的算法会考虑这些信息并使推荐的电影倾向于用户的偏好。

### 9.3.2 批处理和实时阶段

推荐算法处理9.3.1节中提到的 ICM 和 URM 矩阵，他们通过网络服务接口与机顶盒服务端对接，如图9.3所示。

为了满足严格的实时需求，推荐系统以及相应算法采用基于模型的方式[33, 9]，例如，他们首先建立关于用户评分和物品的模型，然后计算推荐结果。因此，算法从逻辑上

被划分成两个阶段：批处理阶段和实时阶段。

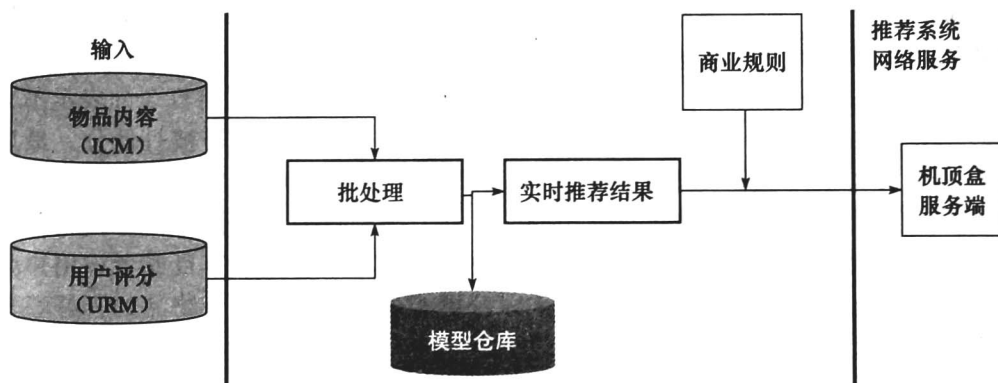


图 9.3 推荐系统：批处理和实时阶段

- 批处理阶段构造输入数据的低维表示(模型)。它一般在非服务高峰期运行，执行频率取决于新物品/用户加入系统的比例(如一天一次)。
- 实时处理阶段利用模型处理来自 Web 服务接口的访问并且满足实时性的限制。系统的输出结果可以进一步被后处理阶段、市场运营规则(如推广某些电影，过滤一些频道)所约束。

模型库使得这两个阶段是可以异步的，比如，当实时阶段利用一个已有模型为用户推荐时，批处理阶段可以同时计算一个新的模型。

推荐过程即使有了这样逻辑的划分，由于输入数据的大小、相关时间以及内存需求等要求，在实际环境中模型构建仍是一个挑战性的工作。基于这个原因，我们实现了高性能的、并行的高要求矩阵运算，并对大而稀疏的矩阵进行了优化。这些运算包括矩阵—矩阵和矩阵—向量相乘，矩阵转置，行/列正交以及奇异值分解(svd)。特别地，三个推荐算法中有两个(一个基于内容的算法，一个协同过滤算法)应用了 svd，大大降低了空间维度，在空间复杂度和时间复杂度方面都有所提升。在后面章节可以看到，仅应用 svd 定义了一个数据模型，它清理了数据噪声并增强了相似信息间的相关性。

实际数据集包含百万级的用户和物品，理论上他们的内存需求量极高。幸运的是，URM 和 ICM 等矩阵是稀疏的。实际上，与全部的记录大小相比大部分用户只评价/观看了一小部分物品(比如，在节目单数千套电影中，用户只看了其中一小部分)。稀疏矩阵可以使用非常有效的方式来处理。注意，即使这些矩阵是稀疏的，在内存中维护这些数据依然是困难的。由于这个原因，我们选择一个基于一种内存虚拟化(类似于操作系统的交换能力)的解决方案。不同于操作系统的虚拟内存，为了限制内存与存储设备间的数据交换，我们的虚拟化策略是针对每个矩阵的具体操作进行定制的。

## 9.4 推荐算法

该推荐系统实现了一个基于内容的算法(CB)和两个协同过滤算法(CF)。

- 基于内容隐语义分析的算法，即 LSA-CB。
- 基于物品的协同过滤算法，即 item-based-CF。
- 基于降维的协同过滤算法，即 SVD-CF。

以下章节我们简要介绍推荐算法。9.4.2 节~9.4.4 节分别详细讨论了 3 种算法的实现细节：LSA、item-based 算法和降维算法。



### 9.4.1 推荐算法概述

推荐算法可以归类为基于内容的算法和协同过滤算法。

基于内容(参考第3章)的推荐方法根源于信息检索,提供了许多工具完成文本信息检索,如文档、网站、新闻以及邮件消息。

一个基于内容的系统是基于对物品内容的分析。物品的模型是由一组代表其内容的特征组成的。基于内容技术的隐含假设是物品具有的意义以及物品间的相关性可以总结为以下特点:

- 物品的每个特征被赋予一个权值来表示其对物品的表达能力;
- 相似的物品具有相似的特征;
- 包含一个特征的物品越多,该特征的表达能力越弱(例如,在区别该物品与其他物品时不够明显)。

特征抽取是这类系统最关键的部分,它在IPTV中非常有挑战性,其对应的资源是非文本的,如音频/视频流。例如,一部电影的文本特征可能是类型(如喜剧)、演员表等。更多有趣的信息可以通过分析音轨得到,该技术[11]是最近的技术,因此有必要调查一下该技术是否会给IPTV领域带来一定的提升。

物品在基于内容的推荐系统中,典型的表示方法是词袋法(BOW)[6],该方法考虑文本特征并且我们只保留词频而忽略任何语法/语义关系。通常这些词会经过分词、去除停用词以及词干化[32]等预处理。前者将文本简单地分成标记(如单词)。在特定领域下不能有效表示物品的标记(停用词)会被忽略。最后,词干化通常用于通过将标记转化为其形态上的词根来对一些语法变形进行标准化。比如,单词“play”“player”“playing”和“played”会被转化为它们的词根形式“play”。预处理完成后,每个标记被赋予一个权值,该权值与词频成比例,词频可以通过不同的方法标准化,而最常用的方法是TF-IDF[26, 32]。

我们将BOW表示结果记作矩阵 $W$ ,其中每一列 $i$ 表示物品 $i$ 而元素 $w_{ic}$ 表示物品 $i$ 的元数据 $c$ 的权重(相关性)。元数据 $c$ 可以是电影类型、演员或导演,也可以是电影梗概中抽取的标记。我们将在9.4.2节讨论如何处理不同类型的元数据。类似地,用户模型也可以表示为标记空间中的向量。实际上,用户模型可以由用户评过分的物品对应的线性组合向量进行表示,权值为用户给相应物品的评分。通过比较表示用户模型的向量以及表示物品的向量的相似度进行推荐。最相似的物品被推荐给用户。向量间的相似度可以由多种距离公式表示,如欧氏距离和余弦距离[26]。

基于内容的推荐系统[1, 3, 21]通过计算物品特征,把与用户以前喜欢过的物品相似的物品推荐给该用户。例如,系统分析用户过去喜欢的电影从而为该用户建立一个模型,模型特征为该用户喜欢的演员、制片人、电影类型和导演等。然后,与该用户偏好相似度较高的电影被推荐给该用户。比如,如果一个用户过去看了许多动作片,那么他会被推荐其他的动作片。基于内容的推荐系统的这种特性有2个直接的影响:它能够确保被推荐的物品符合用户的兴趣,但是,同时这些被推荐的物品是显而易见的并且种类比较单一。这个问题通常被称为过度个性化问题[3]。

基于内容的技术的主要优点是它们基于明确的特征,因此可以为被推荐的物品提供直接、合理的解释,更重要的是,基于内容的过滤是基于一种著名的、成熟的技术。

与基于内容的推荐系统相比,协同过滤推荐系统(参见第5章)尝试基于其他用户的偏好[29, 1]向指定用户推荐商品。实际上,在日常生活中,我们依赖别人的推荐,如口口

相传或电影评论。这类系统利用集体用户的观点帮助个人有效地找到他们感兴趣的内容。协同推荐系统支持并增强这个过程。它们基于以下 2 个假设：

- 许多具有相似品味的用户形成不同的组，他们对物品的评价也相似；
- 相关的物品被同一组的用户的评价是相似的。

相关性的概念与物品间基于内容的相似性是非常不同的。比如，我们说一个电影不论它的内容是什么，它和另一部电影是相关的因为用户社区对这两部电影的评价相同。比如，如果一个用户看了电影《阿甘正传》，那么从协同推荐系统的角度看应该推荐他看《第六感》。这两部电影显然是没关系的，因为从内容上看它们是不相似的，但是由于大部分看了电影《阿甘正传》的人也看了《第六感》，他们实际上是强相关的。

基于以上 2 个假设，我们可以定义 2 类协同推荐系统，分别是基于用户的协同推荐和基于物品的协同推荐[34]。这 2 类推荐系统都是基于社交互动的。实际上，由于性能较差以及内存和时间的限制，基于用户的协同推荐系统很少使用。

注意，协同推荐不需要抽取物品的任何特征。因此，这类系统没有基于内容推荐系统所具有的缺点。特别是，由于协同推荐是基于其他用户的偏好，它能够适用于任何内容。更重要的是，它能够推荐任何物品，即使这些物品的内容与之前喜欢的物品的内容完全不相干。

然而，协同推荐也有它们自身的限制。最主要的缺点是协同推荐受到首次评分问题的影响。既然为当前用户推荐与他喜欢的物品最相关的物品，那么新物品就得不到推荐，因为没有用户为它评价同时系统无法为该物品定义一个模型。因此，直到物品得到足够多的用户评价，系统才能推荐它们。由于这种原因，协同推荐算法不适用于直播电视领域，因为在该领域中新节目加入系统的比率很高，并且这些节目只能在受限的时间内（如几小时）展现并得到评价。注意，基于内容的推荐不存在这个问题，因为新节目加入节目集后可以通过它们的特征建立模型。

第二个问题是稀疏性问题。实际上，协同推荐的有效性依赖于具有相似偏好的用户集合的大小。遗憾的是，在任何推荐系统中与要估计的评价相比已获得的评价是非常少的。因此，不太可能为只有单一兴趣的用户推荐，因为没有其他用户与他足够相似的。

以上两点导致的结果是，在活动开始阶段，新系统无法提供精确的推荐，这个称为冷启动问题。所有的推荐系统，包括基于内容推荐和协同推荐系统，都存在这个问题，然而这个问题对于协同过滤推荐系统更加明显，因为它是只基于用户评价的。

另外，由于热门物品最容易得到用户评分，所以推荐系统更倾向于推荐这些最热门的物品。例如，如果一部电影只被很少的用户评分，那么它将很难获得推荐，因为对它的预测评分可能是不可靠的。

#### 9.4.2 基于内容隐语义分析算法

Fastweb 中实现的基于内容的推荐算法是基于 BOW 的方法，该方法采用隐语义分析来提高推荐算法的可靠性。

参考图 9.2，节目目录中每个物品特征的检索是通过数据搜集器完成的。这些特征经过过滤和加权后形成 ICM。一个物品的特征被分类到多个组，称为元数据。不同种类的物品具有不同的元数据集：

- 视频点播内容：演员表、导演、制片人、标题、系列标题、插曲名称、工作室名称、国家、年份、上映时间、内容摘要、语种；

- 直播 IPTV 节目：演员表、导演、制片人、频道和时间表、国家、播放时间、年份、内容摘要。

视频点播内容与 IPTV 直播内容两者之间的最大不同是前者可以在任何时间根据用户需求访问，而后者只能在某个时间段某个电视频道上观看。推荐系统必须考虑这一点。

对每个物品，元数据被表示为字符串(如标题)或字符串向量(如演员表)。根据元数据的种类，每个字符串被分别进行不同的预处理和加权。如果元数据包含固有名称(如演员、导演等)，我们不进行任何处理而是保留原字符串。另一方面，包含句子(如标题和内容摘要)的元数据被分词、过滤(去除停用词)以及词干化。此外，有些元数据比其他的元数据更重要，那么它们会被赋予更高的权值。通过交叉验证方式我们得到所有种类元数据(如标题、内容摘要、演员表、导演)的权值。此外，内容摘要中每个词干的权值被乘上相应的 TF-IDF 值[26, 32]。

比如，我们考虑一部包含以下元数据的电影：

- 标题：“the title”。
- 电影类型：“comedy”。
- 演员表：“FirstName1 LastName1”，“FirstName2 LastName2”。
- 内容简介：“The movie's plot”。

ICM 矩阵  $W$  相关的列的以下相应元素包含非空权值：

- titl;
- 电影类型;
- FirstName1-LastName1;
- FirstName2-LastName2;
- Movi;
- Plot。

其中，演员和类型保持不变而内容摘要和标题被分词、词干化(“titl”是“title”的词干)以及去除停用词(如“the”)。

除了数据预处理，基于内容的算法性能也由于 LSA(隐语义分析)得以提高。LSA 一种在文档自动索引和检索等[16, 8]信息检索领域非常著名的方法。这种方法利用了词汇和文档的隐式结构(隐语义)。该技术表现为将矩阵  $W$  分解为正交的因子集合，通过这些因子的线性组合可以近似表示原矩阵。矩阵分解过程通过奇异值分解(SVD)完成。

假设 ICM 是一个  $c \times n$  的矩阵( $c$  个元数据和  $n$  个物品)，它可以被分解成 3 个矩阵  $U(c \times l)$ ， $S(l \times l)$  以及  $V(n \times l)$ ：

$$W \approx USV^T \quad (9.2)$$

其中  $l$  是物品隐语义特征个数。一般而言， $l$  是未知的并且它必须通过交叉验证等技术计算得到。 $S$  包含矩阵  $W$  的前  $l$  个最大的特征值，可以粗略地认为这些特征值表示每个隐特征的重要性。 $U$  和  $V$  的列是正交的并且分别表示为左右奇异向量。根据 Frobenius 范数[25]内积是矩阵  $W$  最好的  $l$  阶近似。注意，不考虑三个结果矩阵行和列的线性组合 SVD 是唯一的并且按照惯例， $S$  的对角线元素是正的并按照值降序排列。

SVD 定义了一个新的向量空间，它的维度不是  $c$  个元数据而是  $l$  个隐语义特征，其中  $l$  远小于  $c$  ( $l \ll c$ )。我们可以通过对矩阵  $W$  的相关列进行投影(折叠)从而在隐空间中表示物品  $i$ ，对列向量  $d_i$  的投影表示为

$$\tilde{d}_i = U^T d_i \quad (9.3)$$

同样, 元数据  $c$  可以通过对矩阵  $W$  相关行的投影在隐空间中表示,  $W$  中投影的行  $w_c$  记作

$$\tilde{w} = w_c VS \quad (9.4)$$

图 9.4 描述了折叠过程。可以看到我们也可以将向量投影到原本的空间, 得到原向量的近似表示。尽管 LSA 是原 BOW 空间的近似表示, 它有 2 个主要的优点:

- 它在内存以及计算复杂度上带来了很大的提升。事实上, 一旦 SVD 在批处理阶段计算完成, 系统可以在比 BOW 空间更小的 1 维隐语义定义的低维空间中进行实时运算。
- 通过保留最重要的 1 个特征, 我们去除了噪声并增强了物品与元数据间的关系。比如, 如果 2 个元数据同时出现在多个物品中, 这意味着他们是相关的, 那么它们在隐语义空间中同样表达类似的关联性。由于存在隐式依赖[31, 16], 这些相关性可能是间接的。

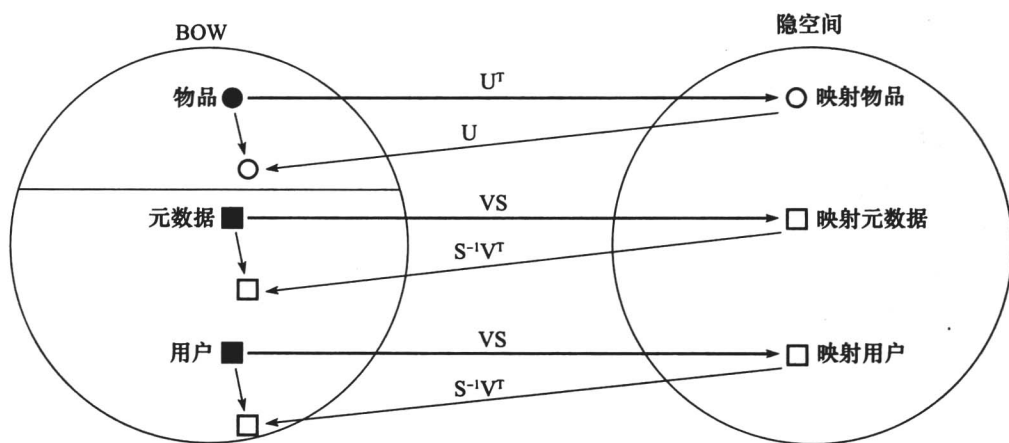


图 9.4 用户、物品及元数据向隐空间映射过程

SVD 的主要问题是其计算复杂度: 通常情况下, 一个  $m \times n$  的矩阵分解的复杂度是  $O(mn^2)$ 。但是, 对于稀疏矩阵, 则存在高效的、可扩展的解决方案。比如, Lanczos[5]实现的 SVD 算法对稀疏、大规模矩阵进行了优化, 用  $z$  表示 URM 中非零元素的个数, 则空间复杂度为  $O(z)$ , 计算复杂度为  $O(zl)$  (比如, 与  $z$  以及要计算的特征值[36, 28]个数成比例)。Fastweb 推荐系统采用了 SVD 的 Lanczos 实现, 能对接到多处理器架构上运行。

向用户推荐物品需要计算它们的相关性(评价)。因此, 我们不仅将物品表示在隐空间中, 也把用户表示在相同的空间中, 这样我们就可以计算用户—物品相关性。用户表示为一系列评分, 也可以是 ICM(如一个元数据)的一行, 用户评分可以通过式(9.4)的方式映射到隐空间, 式中  $w_c$  必须被替换为用户特征(比如, 用户打分的行向量)。一旦将物品和用户在同一个向量空间中表示, 我们就可以采用任一种向量间相关性度量方式计算物品  $i$  对用户  $p$  的相关性, 记作  $\hat{r}_{pi}$ 。Fastweb 采用的度量方式是压缩的余弦距离。假设  $l$  维向量  $\tilde{r}_p$  和  $\tilde{a}_i$  分别表示用户和物品的投影, 则用户  $p$  对物品  $i$  的评分估计值为

$$\hat{r}_{pi} = \frac{\sum_{e=1}^l \tilde{r}_{pe} \cdot \tilde{a}_{ie}}{\sqrt{\sum_{e=1}^l [\tilde{r}_{pe}]^2} \cdot \sqrt{\sum_{e=1}^l [\tilde{a}_{ie}]^2} + \gamma} \quad (9.5)$$

其中,  $\tilde{r}_{pe}$  表示向量  $\tilde{r}_p$  的第  $e$  个元素。常量  $\gamma$  是压缩因子, 用于修正缺失信息时的距离,

比如,当用户或物品向量非常接近原点(比如,物品只有非常少的元数据)而变得无意义。

这种表示方式也支持结合用户的显式偏好,比如,用户明确表示喜欢的演员。实际上,用户显式偏好向量可以看作与物品的元数据向量相似。一旦显式偏好被投影到隐空间,投影后的用户信息与显式偏好信息可以结合起来表示用户图谱,该图谱偏向于显式偏好。

### 9.4.3 基于物品的协同过滤算法

基于物品的协同过滤算法[9, 27]采集物品间的基本关系。根据 9.4.1 节的解释,若用户群体认同它们的评分,则两个物品是相似的(基于“协同”的角度)。这种相似性可以表示为一个  $m \times m$  的矩阵,记作  $D$ ,其中元素  $d_{ij}$  表示物品  $i$  和物品  $j$  之间的相似性。注意, $D$  可能是不对称的(比如,文献[9]中描述的基于条件概率的相似性),即  $d_{ij} \neq d_{ji}$ 。那意味着,物品  $i$  可能非常相似于物品  $j$ (因此,如果一个用户喜欢物品  $i$  那么他也会喜欢物品  $j$ ),即使物品  $j$  不相似于物品  $i$ 。

基于物品的算法将物品表示在用户评分空间中,即一个物品是一个向量,其维度是  $n$  个用户的评分。每一维的元素是用户的打分值。因此,物品  $i$  对应于矩阵  $R$  中的第  $i$  列并且物品间的关系由相关的向量的相似性表示。在以下章节介绍多个计算向量相似性的技术。

根据图 9.3 表示的系统架构,矩阵  $D$  表示推荐系统模型并且其计算(密集型计算)被放在系统批处理部分进行。实时部分使用该模型生成推荐列表。给定一个要推荐的用户特征(表示为评分向量),我们可以通过计算用户对与物品  $i$  相似的物品的评分进行加权求和来预测他对物品  $i$  的评分。这些评分由与物品  $i$  的相似度加权。 $Q_i$  表示与物品  $i$  相似的物品集合,预测  $\hat{r}_{pi}$  的计算公式如下:

$$\hat{r}_{pi} = \frac{\sum_{j \in Q_i} d_{ji} \cdot r_{pj}}{F} \quad (9.6)$$

其中  $F$  是标准化因子。该因子可以简单地设为 1 或根据文献[27],它可以如下计算:

$$F = \sum_{j \in Q_i} |d_{ji}| \quad (9.7)$$

以此确保  $\hat{r}_{pi}$  在预定的评分范围内。注意,作为基于物品方法以及基于模型方法,即使用户  $p$  在模型创建(实际上是计算物品模型的批处理阶段)期间没有被考虑,他也能够被推荐。这就允许,1)使用用户的部分样本(比如,为了满足时间和内存的限制)建立模型;2)为用户做推荐是基于计算好了的模型,即使此刻他的特征相对于模型是新的或是被更新了的。

一旦计算好了那些数据集中没有被目标用户评分的物品的预测评分,这些预测值会被排序并且分值最高的  $N$  个物品组成 top- $N$  推荐列表。

如果我们只考虑相似度大于阈值的物品或最相似的  $k$  的物品, $Q_i$  集合会减小。后一种方法是经典的  $kNN(k$  近邻)方法。根据 9.6 节所讲,改变  $k$  值,推荐质量会相应地改变。

当使用隐式数据集时,相似度度量是通过基于频数的方法计算的,该方法在文献[9]中由 Deshpande 和 Karypis 论述。比如,我们仅处理二元值,物品  $i$  与物品  $j$  之具有较高的相似度意味着当有人购买了物品  $i$  时,他很可能也会购买物品  $j$ 。

我们可以将每个物品看作用户评分空间(该空间坐标是二元值的)的一个向量来处理隐式数据。同样,两个物品间的相似度可以通过计算相应的向量得到,如计算余弦距离。



对于隐式评分,余弦相似度只是一个通用的方法的特例,该方法在后面称作直接关系(DR)。在其基本公式中,与DR一起使用的物品-物品矩阵  $D$  为

$$D = R^T \cdot R \quad (9.8)$$

主对角线上元素  $d_{ii}$  是物品  $i$  被评分的总数,而其他元素  $d_{ij}$  表示同时浏览物品  $i$  和物品  $j$  的用户总数。模型(如  $D$ )可以通过后标准化方式进行后处理,一般表达式如下[9]:

$$d_{ij} \leftarrow \frac{d_{ij}}{d_{ii}^\gamma d_{jj}^\beta + c} \quad (9.9)$$

其中,  $\gamma$ 、 $\beta$  和  $c$  是常量,其最优值根据数据集决定。常量  $c$  是压缩因子[14],当获得信息很少是用于修正物品间相似度。

该模型通过使用  $k$ NN( $k$  近邻)方法得以进一步增强。对每个物品,我们只考虑它  $k$  个最相似的物品(看作该物品的邻居), $k$  的取值可以通过交叉验证方式确定。只保留最相似的物品,去除了与目标物品不相关的干扰项,提高了推荐结果的质量。

注意,上面介绍的方法都是基于统计共同打分物品的数量,它可以通过简单的 SQL 语句在 DBMS(数据库管理系统)中高效地完成,而不需要额外的编程实现。

#### 9.4.4 基于降维的协同过滤算法

基于降维技术的协同过滤算法通过有限的特征集合来描述数据(如用户和物品)。这些特征的意义与那些典型的由基于内容的方法抽取的特征不同。事实上,后者是关注物品内容的特征(比如,电影的类型、一首歌的歌手),而协同过滤算法采用的特征不是基于内容而是基于用户群与物品的隐式交互方式。

假设一个物品可以由  $l$  个特征表示,即它可以表示为  $l$  维的特征空间中的一个向量。同样地,一个用户也表示为该空间中的一个向量。因此,用户  $p$  与物品  $i$  间的相关性(如物品与用户兴趣的匹配程度)可以通过计算相应向量的相似度得到,比如,通过它们的内积:

$$\hat{r}_{pi} = \sum_{e=1}^l a_{pe} \cdot b_{ie} \quad (9.10)$$

其中,  $a_{pe}$  和  $b_{ie}$  分别是用户  $p$  和物品  $i$  的第  $e$  个特征。

目标是计算  $l$  个特征值使能最小化估计值  $\hat{r}_{pi}$  和  $r_{pi}$  实际值之间预测误差。

比如,Paterek 在文献[20]中采用了一种优化方法叫作正则化奇异值分解,该方法已经在自然语言处理领域[12]得以应用。用户和物品的  $l$  个特征通过采用基于梯度下降的优化技术最小化 RMSE 估计,每次学习一个特征。RMSE 度量公式定义为

$$RMSE = \sqrt{\frac{1}{n} \sum_{p,i} (\hat{r}_{pi} - r_{pi})^2} \quad (9.11)$$

与 LSA 相似,在实现中再一次使用 SVD,该方法也应用于 URM。实际上,URM 可以如下分解:

$$\hat{R} = U \cdot S \cdot V^T \quad (9.12)$$

其中,  $U$  是  $n \times l$  正交矩阵,  $V$  是  $m \times l$  正交矩阵,  $S$  是包含了前  $l$  个特征值的对角阵,其元素按降序排列。

用户  $p$  对物品  $i$  的评分可以预测为

$$\hat{r}_{pi} = \sum_{e=1}^l u_{pe} \cdot s_{ee} \cdot v_{ie} \quad (9.13)$$

其中,  $u_{pe}$  是矩阵  $U$  中第  $p$  行、第  $e$  列的元素;  $v_{ie}$  是矩阵  $V$  中第  $i$  行、第  $e$  列的元素;  $s_{ee}$  是

矩阵  $S$  中第  $e$  行、第  $e$  列的元素。

假设  $u_p$  表示矩阵  $U$  的第  $p$  行,  $v_i$  表示矩阵  $V$  的第  $i$  行, 式(9.13)可以写作:

$$\hat{r}_{pi} = u_p \cdot S \cdot v_i^T$$

(9.14)

由于矩阵  $U$  和  $V$  的列向量是正交的, 通过式(9.12)两边同乘以  $V$ , 得

$$u_p \cdot S = r_p \cdot V$$

(9.15)

其中, 表示矩阵  $R$ (如用户特征向量  $p$ )的第  $p$  行。因此, 式(9.14)可以重新计算为

$$\hat{r}_{pi} = r_p \cdot V \cdot v_i^T$$

(9.16)

通过式(9.16)我们可以向任何一个用户推荐, 即使他的特征是新的或者模型创建后(如 SVD 已经完成)他的特征被更新了。这种表示大大优于其他降维技术(如规则化 SVD), 其中指定用户的特征在模型创建过程中计算并保持不变。

为了预测用户  $p$  所有的评分, 式(9.16)可以被直接扩展为

$$\hat{r}_p = r_p \cdot V \cdot V^T$$

(9.17)

注意,  $V$  和  $V^T$  的内积结果为一个  $m \times m$  的物品—物品矩阵, 其意义与 9.4.3 节基于物品算法中讨论的物品—物品矩阵  $D$  的意义相似。

与 LSA 相似, 采用这种基于 SVD 的方法有几个优点:

- SVD 在一个低维空间中表示用户和物品。一旦矩阵  $R$  被分解(这个过程会带来很大的挑战), 系统处理的只是  $l$  维的向量, 其维度比  $n$  个用户和  $m$  个物品的原始空间大大减小。
- SVD 去除了数据中的噪声。实际上, 忽略掉较低的特征值, 我们去除了低信息量的数据, 而这些数据很可能是噪声[10, 8]。
- SVD 强化了数据间的关系。因此, 如果 2 个向量(用户或者物品)相似(因为某些关系), 它们在  $l$  维特征空间中的表示比在原空间中离得更近。注意关系可能是间接的, 通过 SVD 可以发现用户或物品间的隐依赖关系。

参考 9.3.2 节描述的算法架构, 矩阵分解式(9.12)由批处理部分完成, 而评分预测式(9.16)由实时处理部分完成。实时处理部分为目标用户对所有未评分的物品估计评分, 然后将评分排序并选择  $N$  个最高的物品形成 top- $N$  推荐列表。在我们的测试中, 通过式(9.16)为指定用户计算 top- $N$  推荐列表花费的时间为几毫秒, 符合实时需求。

9.5 推荐服务

本节描述实现的推荐服务以及它们如何影响用户界面和 IPTV 架构。推荐系统可以产生基于内容的和基于协同过滤的推荐。如图 9.5 所示, 基于内容的推荐可以用于视频点播和电视直播领域, 而协同过滤算法仅用于视频点播。实际上, 在 9.4.1 节我们已经注意到协同过滤算法在新内容不断添加的系统中不适用, 并且只有当新节目被大量的用户观看/评价后协同过滤算法才能推荐。

在当前整合阶段, Fastweb 在实际发布之前将全部推荐服务开放给选定的一部分测试用户。其他用户只能访问其中一部分推荐服务。图 9.6 展示了通过 STB 获得的用户界面的一个样例。

目前提供给所有用户使用的服务只涉及视频点播内容中的一个节目类别。推荐结果由

	VOD	Live TV
基于物品的CF		
SVD-CF		
LSA-CB		

图 9.5 推荐算法在视频点播和电视直播中的应用

9.4.2 节描述的 LSA-CB 算法提供。在系统运行的前几个月更多地采用基于内容的算法，因为根据 9.4.1 节的解释，协同过滤算法会碰到冷启动问题。此外，协同过滤推荐需要记录用户行为。这会使 Fastweb 面临一些微妙的合法性问题，比如，需要获得用户授权以便存储和管理他们的数据，并在确保这些信息机密以及匿名的情况下实现解决方案。



图 9.6 推荐系统用户界面

## 9.6 系统评价

本节首先通过离线测试计算的准确率度量来讨论推荐系统的质量(见第 8 章)。然后，我们介绍推荐系统在线分析中的一些反馈信息。

离线测试是基于搜集到的用户对视频点播中一个节目类别下 7 个月的行为数据。图 9.7 显示了观看数量的演变。观看的平均数量为每天 1600，而周末的时候可以达到 3300。图 9.8、图 9.9、图 9.10 通过分别展示了活跃用户数量、活跃物品数量以及数据集密度随时间的变化过程完成了分析。活跃用户是指那些至少评价一个物品的用户。同样，活跃物品是指至少被一个用户评价的物品。数据集密度是评价的总数与用户数和物品数乘积的比例。从图 9.10 中我们注意到趋势不是单调的。实际上，每当用户看了他的第一部电影，我们就有一个新的活跃用户，同时数据集密度会下降。

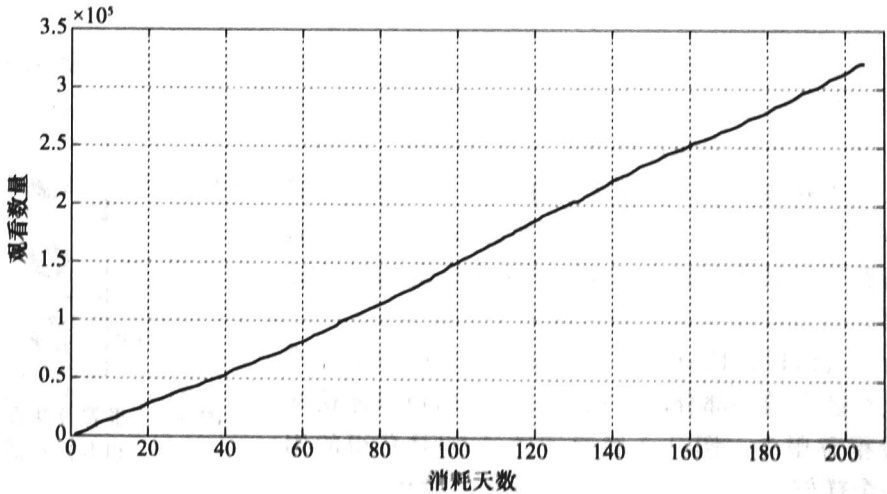


图 9.7 用户关于视频点播中一个类别的 7 个月的行为数据中搜集的观看量

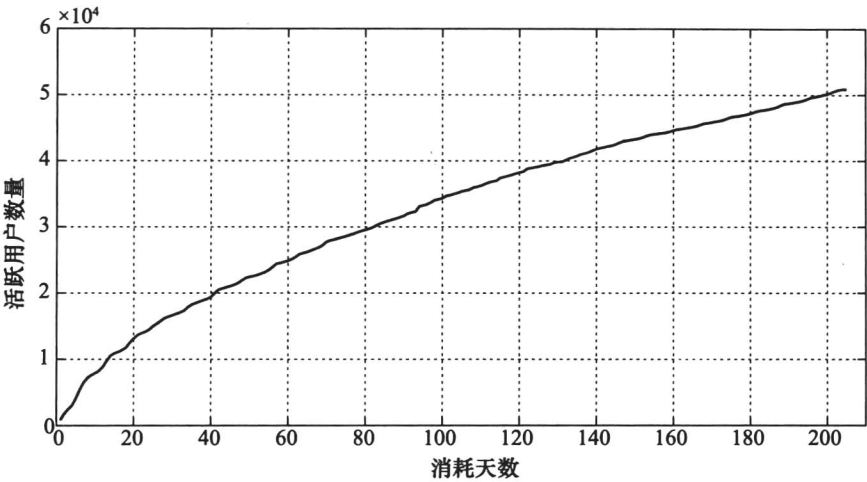


图 9.8 同一视频点播类别中活跃用户数量

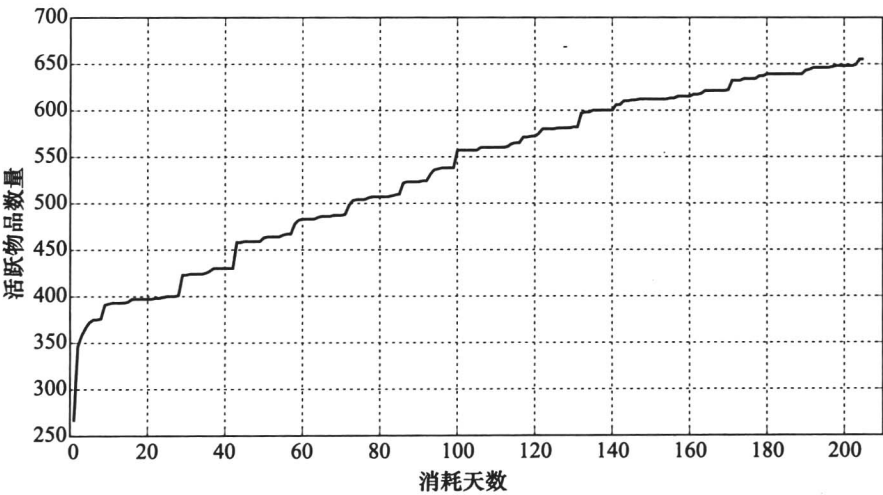


图 9.9 同一视频点播类别中活跃物品数量

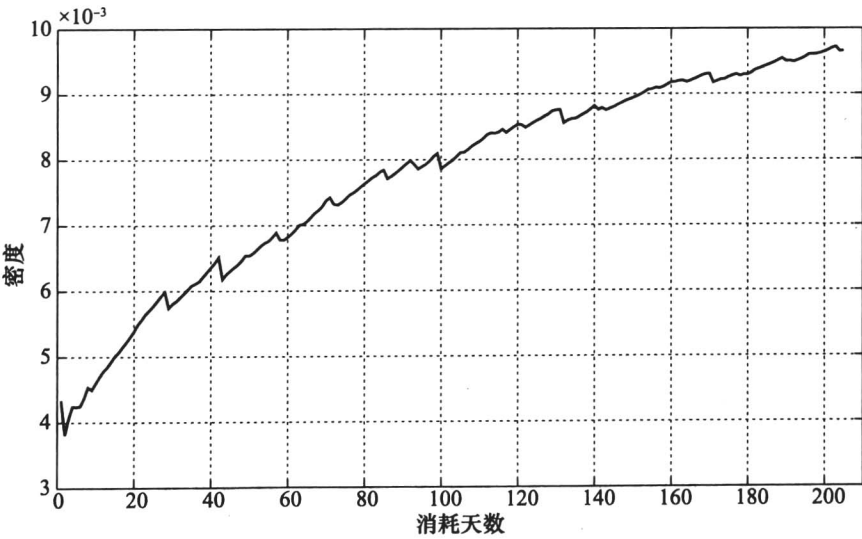


图 9.10 同一视频点播类别中评分密度演变过程。密度是通过考虑观看量(如评价)相对于活跃用户与活跃物品的数量计算得到的

9.6.1    离线分析

推荐系统评价的典型方法是基于误差度量(如 RMSE 和 MAE)[22]或分类精度度量(如召回率、精确度和误检率)[15, 7]。正因为我们只用了隐式评价来表示用户兴趣,实际上我们只能通过精确度指标参数来验证系统的性能及其效果。最后,表 9.1 和表 9.2 表示 9.4.2 节、9.4.3 节和 9.4.4 节描述的三种算法的召回率,算法分别为 LSA-CB 算法,基于物品的协同过滤算法和 SVD 协同过滤算法。

召回率常用于信息检索,它说明了由搜索引擎获取的相关物品的比例。在推荐领域,召回率表示用户有效观看的电影中由推荐系统推荐的数量。为了达到这个目的,我们采用留一法:

- 对测试集中的每个用户,选择一个评价的物品。
- 将选定的物品从用户图谱中删除,并且为修改后的用户图谱生成推荐;过滤出用户已经评价过的物品。
- 如果被删除的物品在我们获得的推荐列表中排在前 5 位,则表示一次命中,比如,用户看过的电影被推荐算法推荐(根据 Fastweb 的用户界面,推荐列表限制为 5 个物品)。
- 对每个用户的每个物品重复以上过程。

召回率表示命中次数相对于测试次数的百分比。

根据不同的算法选择的测试集是不同的。实际上,基于内容的算法通过 ICM 建立模型,那么测试集可以是全部 URM。另一方面,协同过滤算法模型是基于 URM 本身,因此我们通过 10 折交叉验证方法进行了评价,即将用户随机分成 10 份然后其中一份用作测试集用于计算召回率,而剩余的 9 份用于生成模型。每一份测试集通过留一法分析。最终结果是 10 次召回率的平均值。

表 9.1 列出了推荐算法上线 3 个月和 6 个月后的召回率,显示了系统随时间的演变过程。此外,9.4 节描述的 3 个算法的推荐质量是通过一个简单的算法进行比较的,该算法为 top-rated,只用于比较目的。top-rated 算法是一个基本的协同过滤算法,该算法为每个用户推荐固定长度的列表,该列表由高流行度到低流行度进行排序(去除用户已经评价的物品)。

比如,表 9.1 显示在算法上线 6 个月中最好的是基于物品的协同过滤算法,最好的配置为邻居个数  $k$  等于 100。从表 9.1 中我们可以观察到一些特别的方面:

- 在某些情况下,6 个月后的推荐质量比 3 个月低;
- top-rated 算法的质量是比较好的;
- 基于内容的算法推荐质量较差,甚至比 top-rated 算法还差。

关于第一个观察结果,我们期望随着系统搜集评分,我们获得更加精确的用户图谱,从而推荐质量也会提高。然而,事实未必一直如此,比如,根据文献[7]中提及的一系列基于朴素贝叶斯网络(NBN)的基于物品协同过滤算法的实验结果。此外,我们的分析没有

表 9.1    关于指定的视频点播类别的推荐质量

算法	参数	召回率	
		3 个月	6 个月
基于物品的 CF	$k=10$	16.8%	14.9%
	$k=50$	18.7%	16.4%
	$k=100$	<b>19.0%</b>	<b>16.6%</b>
	$k=150$	18.8%	16.5%
SVD-CF	$l=5$	15.1%	12.7%
	$l=15$	12.6%	13.3%
	$l=25$	10.9%	11.5%
	$l=50$	9.3%	9.9%
	$l=100$	6.3%	8.0%
LSA-CB	$l=50$	1.9%	1.7%
	$l=100$	2.3%	2.3%
	$l=150$	2.4%	2.4%
	$l=200$	2.5%	2.5%
top-rated		12.2%	7.7%



考虑评分或物品随时间变化的问题。实际上，如图 9.9 所示，活动 3 个月后有 510 个活跃物品中挑选一个，而 6 个月后有 621 个可供挑选。由于部分反效应，活跃物品在增加，用户评价更多的物品，而算法却忽略了这些物品。无论如何，这最低限度地补偿了物品增加效应。实际上，如图 9.11 所示，随着活跃物品由 510 增加到 621，平均用户图谱长度由每个用户 3 个物品增加到 6 个物品。

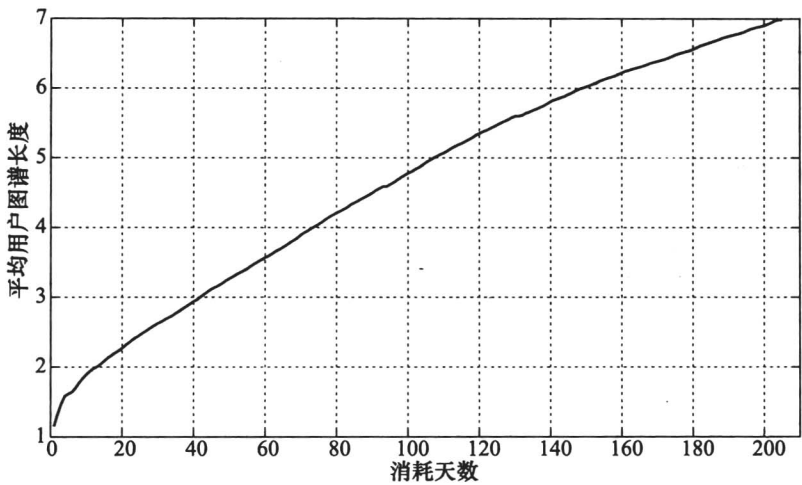


图 9.11 平均用户图谱长度时间演变过程。图谱长度根据视频点播内容类别之一的活跃用户计算

关于第二个和第三个观察结果，他们找到了一个通用的解释。对于基于内容算法的较差的推荐质量以及 top-rated 算法较好的推荐质量，部分是因为基于留一法本身的测试方法。实际上，留一法结果的召回率会偏向于选择最流行的物品算法的召回率，因为它们数量较多，所以更容易被测试。基于内容的算法效果非常差是因为它忽视了物品的流行度。另一方面，top-rated 算法特别有优势，是因为根据文献[7]显示当用户图谱比较短(如冷启动阶段)时，大部分用户更可能观看最流行的电影。此外，用户经常期望推荐系统推荐新奇的东西，而推荐最流行的物品不能解决惊喜度[15]的问题。

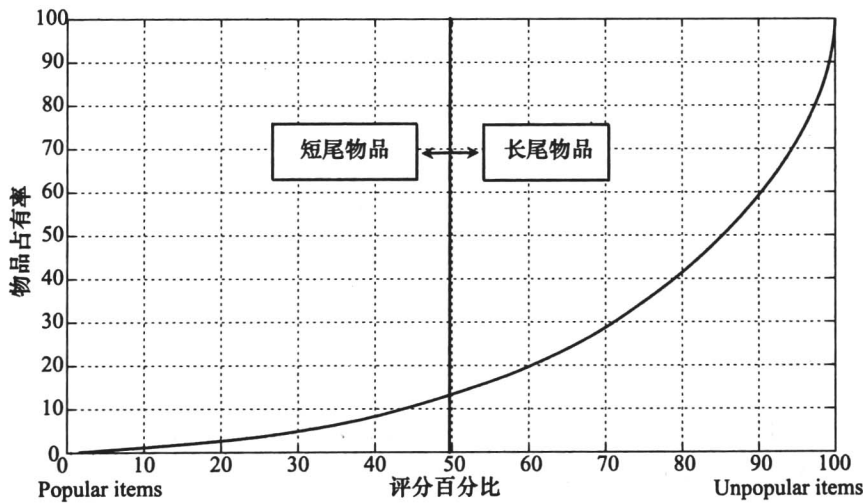


图 9.12 长尾效应：50%的评价集中于10%~12%的最流行物品(短头)

由于以上原因，我们在接下来更深层的推荐算法质量评价中，去除了最流行的物品并且只根据非流行物品计算召回率，以应对著名的长尾[2]问题。图 9.12 展示了流行物品与

非流行物品的评价分布。比如，我们可以看到 50% 的评价集中于 10% 的最流行的物品(短头)，而其余 90% 的物品(长尾)只占有 50% 的评价：一个实施推荐系统的主要原因是提高长尾物品的销售量，因为它们代表了服务提供商的潜在收入。然而，推荐长尾物品比推荐短头物品难得多。

表 9.2 表示去除了前 10 个最流行的物品后的召回率测试结果(记作 non-top-10)以及去除了短头物品(最流行的，它们代表了总评价的 50%)的召回率测试结果(记作 non-top-50%)。

从表 9.2 中可以得到：

- 基于内容的算法质量保持不变；
- 当推荐非流行物品时，协同过滤算法的质量会下降，而 top-rated 算法是失败的；
- 与基于物品的协同过滤算法相比，非流行物品可以更好地由基于降维的协同过滤算法推荐。

表 9.2 对视频点播内容类别之一中长尾物品的推荐质量，如分别为流行度不在前 10 的物品以及评价不在前 50% 的物品

算法	参数	non-top-10 召回率		non-top-50% 召回率	
		3 个月	6 个月	3 个月	6 个月
基于物品的 CF	$k=10$	14.0%	13.2%	7.7%	9.6%
	$k=50$	<b>14.0%</b>	<b>13.8%</b>	6.8%	9.0%
	$k=100$	13.8%	13.5%	6.2%	8.3%
	$k=150$	13.5%	13.2%	6.1%	7.9%
SVD-CF	$l=5$	6.6%	6.8%	0.7%	1.4%
	$l=15$	11.5%	10.2%	1.2%	3.5%
	$l=25$	12.6%	12.0%	2.2%	4.9%
	$l=50$	11.4%	11.2%	4.8%	7.8%
	$l=100$	7.6%	9.3%	<b>9.8%</b>	<b>11.8%</b>
LSA-CB	$l=50$	2.1%	1.8%	1.8%	1.7%
	$l=100$	2.3%	2.3%	2.0%	2.5%
	$l=150$	2.5%	2.5%	2.1%	2.5%
	$l=200$	2.6%	2.6%	2.2%	2.6%
top-rated		0.4%	1.0%	0%	0%

关于第一个观察结果，基于内容的算法被证实不受物品流行度的影响。

正相反，协同过滤算法的召回率在下降。其中，top-rated 算法的质量急剧下降，而实际上 top-rated 无法推荐长尾物品。

此外，我们可以看到，推荐 non-top-10 时基于物品的协同过滤算法依然是最好的。然而，当我们关注长尾(non-top-50%)时，基于降维的协同过滤算法超越了基于物品的协同过滤算法。同时，我们也可以看到随着系统搜集的评价的增加，基于降维协调过滤算法的推荐质量呈现正趋势，推荐非流行物品的能力也在提高。

9.6.2 在线分析

本节将之前在推荐系统离线分析得到的结果与在线分析(即我们直接在运行中的推荐系统上分析反馈信息)结合起来。根据 9.5 节的解释，结果数据是在视频点播内容的一个类别中实施基于内容的算法后得到的。

为了根据经验评估召回率，我们假设不论用户是在推荐系统推荐之前或之后观看了一部电影，这部电影都是与用户相关的并且代表了推荐系统的一个成功推荐。

我们定义**推荐成功数**，它用于衡量在一个时间段内推荐过的电影被观看的次数。令

$b(t)$ 表示推荐成功数， $w(t)$ 表示用户观看的电影数， $t$ 表示推荐的时间段，则我们可以计算经验召回率为推荐成功数与观看总数的百分比，即

$$\text{经验召回率}(t) = \frac{b(t)}{w(t)}$$

(9.18)

经验召回率表示观看总数中被推荐算法击中的百分比。这个特定的指标依赖于推荐系统为用户服务的时间  $t$ 。请注意，一个很长的时间段  $t$  可能会放宽推荐和观看之间的依赖。表 9.3 列出了通过监测推荐系统 2 小时、24 小时以及 7 天的观看数量计算出的系统平均质量。结果区分了流行的物品与非流行的物品。

从表中可以看出，相对于流行的电影，不流行的电影的经验召回率较高。实际上，即使没有推荐系统的推荐，用户也已经知道了那些流行的电影。比如，用户或者已经看过一部流行的电影(如在电影院)或他根本就不感兴趣。

进一步分析，大约 64%的成功推荐是非流行电影(如 non-top 50%)，而只有 36%是推荐流行电影(如 top 50%)，即推荐系统更鼓励用户观看非流行的电影，这对于长尾是正面效应。

表 9.3 测试的视频点播内容类别的平均经验召回率。结果包含推荐后的 3 个时间段(2 小时、24 小时以及 7 天)，并且结果区分流行物品与非流行物品

	2 小时	24 小时	7 天
全部	17.0%	19.8%	24.7%
top 10	5.1%	7.0%	10.6%
non-top 10	24.2%	27.6%	32.1%
top 50%	9.4%	11.5%	16.2%
non-top 50%	28.4%	32.2%	36.1%

此外，我们通过衡量提升因子(lift factor)来突出推荐系统的优势，提升因子在观看数量中引入，比如，由于使用推荐系统导致观看的增加量。一般来讲，IPTV 系统的观看量依赖用户群的数量。此外，我们也必须考虑与老用户相比，新用户会看更多的电影。除了稳定增加的新用户，我们还有随着市场营销而产生的新用户数激增。图 9.13 表示整个 Fastweb 用户群 2 年多的趋势。陡峭的部分与促销活动有关。由于隐私的原因，真实的用户数量被隐藏而以比例值代替。

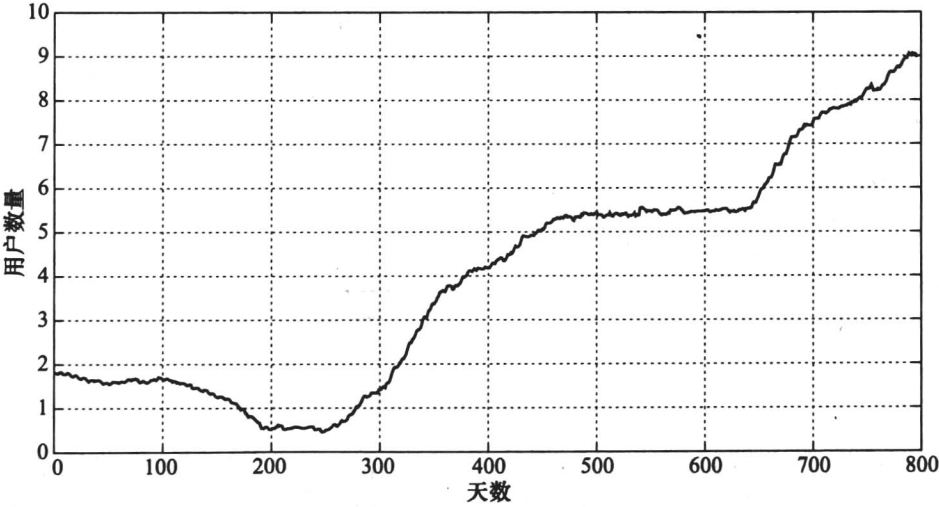


图 9.13 Fastweb 用户数量。真实的数量与列出的值成比例

为了描述用户和观看量的关系，我们定义了一个自回归滑动平均模型(ARMAX)，该模型输入为当前用户群的大小和新用户的数量。ARMAX 模型参数根据整合 ContentWise 前 50 周的用户活动数据进行估计和验证。图 9.14 比较了真实的观看数和模型估计的观看数。为了平滑每天的变化，观看数根据周进行聚合。将数据分成训练集和验证集，验证集上 RMSE 结果低于 2%。

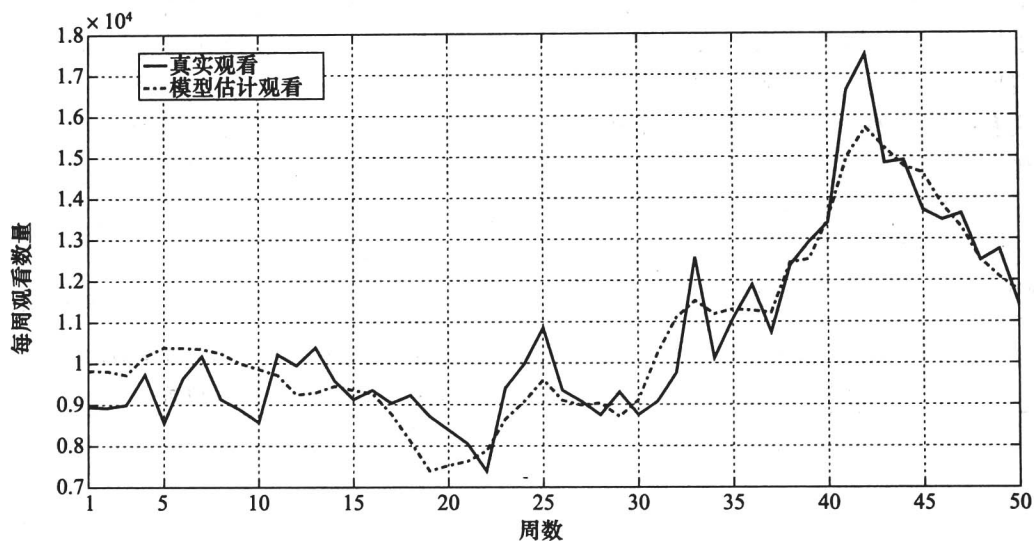


图 9.14 引入 ContentWise 之前每周观看数量的变化

然后该模型用于估计引入推荐系统后前 20 周的观看数量。根据图 9.15，真实值比模型估计的值高，并且这个数字的增加可以归因于推荐系统，因为其他的潜在因素(比如，市场推广活动)也引入 ARMAX 模型中。平均来说，这一段时间的提升因子为 15.5%。

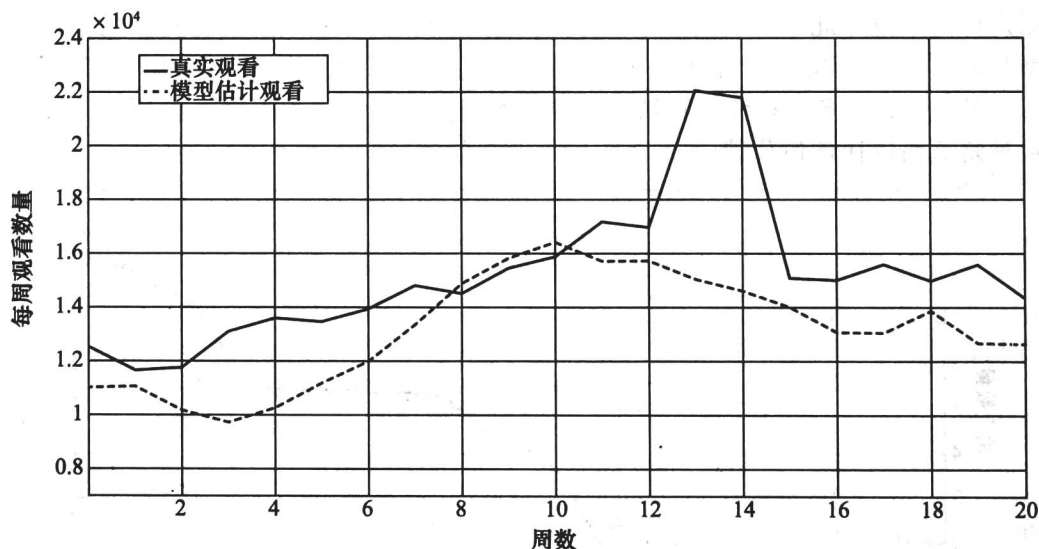


图 9.15 引入 ContentWise 后每周观看数量的变化

最后，我们分析用户如何在选定的视频点播内容类别中寻找感兴趣的内容。图 9.16 显示了每天分别通过推荐系统，基于关键字的搜索引擎以及字母表浏览的搜索请求数量。推荐系统和其他搜索工具的请求差距说明用户有效地使用推荐算法来查找电影。

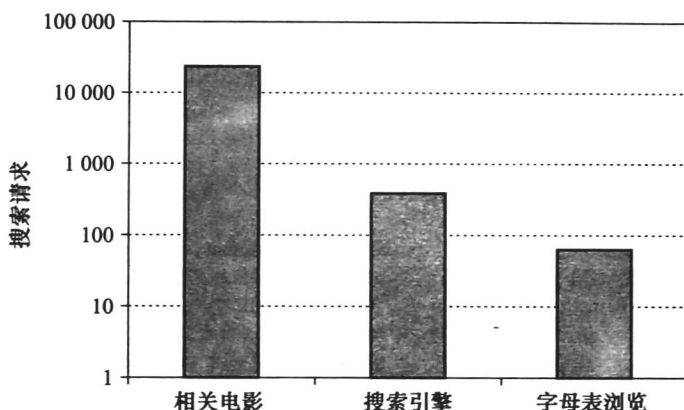


图 9.16 不同搜索感兴趣内容的途径的比较：推荐系统(相关电影)、基于关键字的搜索引擎以及字母表浏览。图中值为取对数后的结果

## 9.7 总结

将 ContentWise 推荐系统整合入 Fastweb 架构对用户和服务提供商都产生了积极的影响。源于在线分析中的三个主要方面证实了推荐系统的正面影响：1) 用户更喜欢通过推荐系统界面浏览视频点播内容类别；2) 在较短的时间内，用户会接受并观看系统推荐的电影；3) 用户观看的电影数有所增加。

为了全面发布推荐服务，当前正在通过测试和调整所实现的推荐算法并且监测系统冷启动阶段，对 Fastweb 的其他类别进行进一步实验。其他正在进行的工作还包括应对从用户行为中估计隐式评分的精度问题。

## 参考文献

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on* **17**(6), 734–749 (2005). DOI 10.1109/TKDE.2005.99
2. Anderson, C.: *The Long Tail: Why the Future of Business Is Selling Less of More*. Hyperion (2006). URL <http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20&path=ASIN/1401302378>
3. Balabanović, M., Shoham, Y.: Fab: content-based, collaborative recommendation. *Commun. ACM* **40**(3), 66–72 (1997). DOI <http://doi.acm.org/10.1145/245108.245124>
4. Bell, R.M., Koren, Y.: Scalable collaborative filtering with jointly derived neighborhood interpolation weights. *7th IEEE Int. Conf. on Data Mining* pp. 43–52 (2007)
5. Berry, M.W.: Large-scale sparse singular value computations. *The International Journal of Supercomputer Applications* **6**(1), 13–49 (1992). URL [citeseer.ist.psu.edu/berry92large.html](http://citeseer.ist.psu.edu/berry92large.html)
6. Chai, K.M.A., Chieu, H.L., Ng, H.T.: Bayesian online classifiers for text classification and filtering pp. 97–104 (2002). DOI <http://doi.acm.org/10.1145/564376.564395>
7. Cremonesi, P., Lentini, E., Matteucci, M., Turrin, R.: An evaluation methodology for recommender systems. *4th Int. Conf. on Automated Solutions for Cross Media Content and Multi-channel Distribution* pp. 224–231 (2008)
8. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. *Journal of the American Society of Information Science* **41**(6), 391–407 (1990). URL <http://citeseer.ist.psu.edu/deerwester90indexing.html>
9. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)* **22**(1), 143–177 (2004). DOI <http://doi.acm.org/10.1145/963770.963776>
10. Furnas, G.W., Deerwester, S., Dumais, S.T., Landauer, T.K., Harshman, R.A., Streeter, L.A., Lochbaum, K.E.: Information retrieval using a singular value decomposition model of latent



- semantic structure. pp. 465–480. ACM Press, New York, NY, USA (1988). DOI <http://doi.acm.org/10.1145/62437.62487>
11. Geneve, U.D., Marchand-maillet, S.: Vision content-based video retrieval: An overview
12. Gorrell, G.: Generalized Hebbian Algorithm for Incremental Singular Value Decomposition in Natural Language Processing. 11th Conference of the European Chapter of the Association for Computational Linguistics (2006)
13. Hand, S., Varan, D.: Interactive narratives: Exploring the links between empathy, interactivity and structure pp. 11–19 (2008)
14. Herlocker, J., Konstan, J., Riedl, J.: An algorithmic framework for performing collaborative filtering. 22nd ACM SIGIR Conf. on R&D in Information Retrieval pp. 230–237 (1999)
15. Herlocker, J., Konstan, J., Terveen, L., Riedl, J.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) **22**(1), 5–53 (2004)
16. Husbands, P., Simon, H., Ding, C.: On the use of singular value decomposition for text retrieval (2000). URL [citeseer.ist.psu.edu/article/husbands00use.html](http://citeseer.ist.psu.edu/article/husbands00use.html)
17. Jensen, J.F.: Interactive television - a brief media history **5066**, 1–10 (2008)
18. Kelly, D., Teevan, J.: Implicit feedback for inferring user preference: a bibliography. SIGIR Forum **37**(2), 18–28 (2003). DOI <http://doi.acm.org/10.1145/959258.959260>
19. Lee, Y., Lee, J., Kim, I., Shin, H.: Reducing iptv channel switching time using h.264 scalable video coding. Consumer Electronics, IEEE Transactions on **54**(2), 912–919 (2008). DOI [10.1109/TCE.2008.4560178](http://doi.org/10.1109/TCE.2008.4560178)
20. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. Proceedings of KDD Cup and Workshop (2007)
21. Pazzani, M., Billsus, D.: Content-based recommendation systems. The Adaptive Web: Methods and Strategies of Web Personalization, Lecture Notes in Computer Science pp. 325–341 (2006)
22. Powers, D.M.W.: Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness and Correlation (2000)
23. Rafey, R.A., Gibbs, S., Hoch, M., Gong, H.L.V., Wang, S.: Enabling custom enhancements in digital sports broadcasts pp. 101–107 (2001). DOI <http://doi.acm.org/10.1145/363361.363384>
24. Rokach, L., Maimon, O., Averbuch, M.: Information Retrieval System for Medical Narrative Reports, Lecture Notes in Artificial intelligence 3055, page 217–228 Springer-Verlag (2004)
25. Saad, Y.: Numerical methods for large eigenvalue problems. Halsted Press New York (1992)
26. Salton, G. (ed.): Automatic text processing. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1988)
27. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. 10th Int. Conf. on World Wide Web pp. 285–295 (2001)
28. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of Dimensionality Reduction in Recommender System-A Case Study. Defense Technical Information Center (2000)
29. Schafer, J., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. pp. 291–324 (2007)
30. Sun, J., Gao, S.: Iptv based on ip network and streaming media service station. MIPPR 2007: Remote Sensing and GIS Data Processing and Applications; and Innovative Multi-spectral Technology and Applications **6790**(1), 67904Q (2007). DOI [10.1117/12.749611](http://dx.doi.org/10.1117/12.749611). URL <http://link.aip.org/link/?PSI/6790/67904Q/1>
31. Valle-Lisboa, J.C., Mizraji, E.: The uncovering of hidden structures by latent semantic analysis. Inf. Sci. **177**(19), 4122–4147 (2007). DOI <http://dx.doi.org/10.1016/j.ins.2007.04.007>
32. Van Rijsbergen, C.J.: Information Retrieval, 2nd edition. Dept. of Computer Science, University of Glasgow (1979). URL [citeseer.ist.psu.edu/vanrijsbergen79information.html](http://citeseer.ist.psu.edu/vanrijsbergen79information.html)
33. Vozalis, E., Margaritis, K.: Analysis of recommender systems algorithms. Proc. of the 6th Hellenic European Conf. on Computer Mathematics and its Applications (2003)
34. Wang, J., de Vries, A.P., Reinders, M.J.T.: Unifying user-based and item-based collaborative filtering approaches by similarity fusion. pp. 501–508. ACM Press, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1148170.1148257>
35. Zhang, H., Zheng, S., Yuan, J.: A personalized tv guide system compliant with mhp. Consumer Electronics, IEEE Transactions on **51**(2), 731–737 (2005). DOI [10.1109/TCE.2005.1468026](http://doi.org/10.1109/TCE.2005.1468026)
36. Zhang, X., Berry, M.W., Raghavan, P.: Level search schemes for information filtering and retrieval. Information Processing and Management **37**(2), 313–334 (2001). DOI [http://dx.doi.org/10.1016/S0306-4573\(00\)00032-7](http://dx.doi.org/10.1016/S0306-4573(00)00032-7)

# 走出实验室的推荐系统

Jérôme Picault、Myriam Ribière、David Bonnefoy 和 Kevin Mercer

**摘要** 从提取数据到将推荐结果展现给用户，个性化系统是一个由很多交互模块组成的复杂系统。对于这样一个系统的每一个部分，都有大量的方法、工具、算法和途径，如多种多样的数据处理方法和元数据处理方法、用户模型、过滤技术、精度指标、个性化程度。不仅如此，现实环境中的推荐系统是很难操控的，更大、更复杂环境的一部分——通常推荐系统只是一个更大的应用环境的一部分，而外部环境因素对推荐系统的设计提出了更多限制条件。例如，数据格式不规范，或者外部环境系统带来的一些架构方面或隐私方面的限制条件。这些现实情况给推荐系统的设计带来更大的难度，并且很容易导致失误。基于作者的经验和对其他类似项目的研究，本章旨在对个性化推荐系统的设计、实现和评估提供指导。结合实际经验，本章介绍了推荐系统在设计之初就应该考虑的方方面面，如何避免误区。本章提出了设计推荐系统时需要考虑的主要因素，并通过对实际案例的研究对那些主要因素进行了说明，以便在必须面对许多复杂的抉择时协助操作。

## 10.1 简介

从提取数据到将推荐结果展现给用户，个性化系统是一个由很多交互模块组成的复杂系统。对于这样一个系统的每一个部分，都有大量的方法、工具、算法和途径，如多种多样的数据处理方法和元数据处理方法、用户模型、过滤技术、精度指标、个性化程度等。不仅如此，现实环境中的推荐系统是很难操控的，更大、更复杂环境的一部分——通常推荐系统只是一个更大的应用环境的一部分，而外部环境因素对推荐系统的设计提出了更多限制条件。例如，数据格式不规范，或者外部环境系统带来的一些架构方面或隐私方面的限制条件。这些现实情况使得建立一个推荐系统成为一项艰巨的任务。

本章旨在对个性化推荐系统的设计、实现和评估提供指导。结合实际经验，本章介绍了推荐系统在设计之初就应该考虑的方方面面，以及如何避免误区。

## 10.2 设计现实环境中的推荐系统

现有文献就建设推荐系统需要考虑的诸多方面问题提供了概览。例如，文献[49]列举了驱动个性化推荐系统设计的若干特征和基本原则，如需要考虑内容的特殊性，用户信任和用户参与推荐系统的重要性。文献[25]提出了对协同过滤系统进行深度分析的方法和评估指标，包括推荐系统的用户任务分类方法和关于数据集属性的描述。这篇文章对进行初

---

Jérôme Picault, Alcatel-Lucent Bell Labs e-mail: jerome.picault@alcatel-lucent.com

Myriam Ribière, Alcatel-Lucent Bell Labs e-mail: myriam.riberie@alcatel-lucent.com

David Bonnefoy, Pearltrees e-mail: david.bonnefoy@pearltrees.com

Kevin Mercer, Loughborough University e-mail: K. C. Mercer@lboro.ac.uk

翻译：潘泉波，胡聪(胡户主)，北京邮电大学-刘树栋 审核：张祺，吉林大学-丁彬钊，李青，郑州大学-吴宾

始的技术选型非常有用,如用户建模和算法的选择等。但如何确保在系统设计之初做出明智的选择?这才是此时最需要关心的问题,因为在后续开发过程任何修改都将付出巨大代价。

为了系统地解决这一难题,我们有必要返回上一级,以更大的视角来看系统设计中的主要设计决策是什么以及那些影响设计决策的因素。图 10.1 说明了在本章中提到的方法。

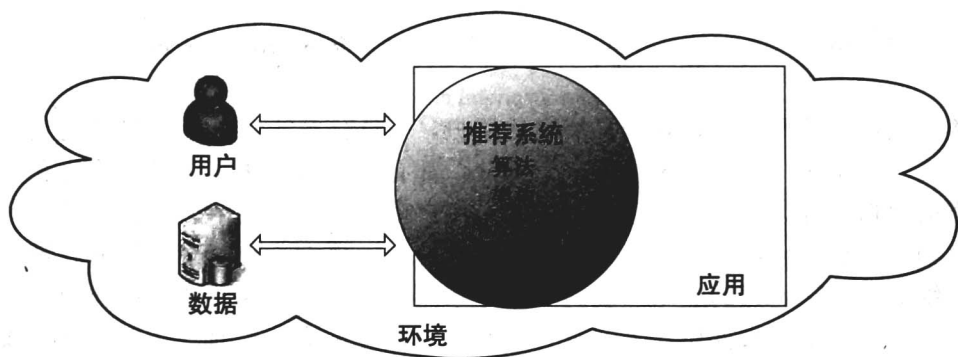


图 10.1 推荐环境中的推荐系统

设计一个推荐系统意味着要在以下几个方面做出抉择:

- 算法: 使用哪些推荐算法?
- 架构: 这个系统如何部署, 集中式还是分布式?
- 用户信息: 用户模型是什么? 用户信息的修正模式是什么?

从大的方面来说, 这些选择受限于推荐系统所处的外部环境系统。因此, 系统地研究推荐系统置身的外部环境是至关重要的。我们建议从以下三个维度来描述:

- 用户: 用户是哪些人, 用户的目的是什么?
- 数据: 具体推荐过程中所使用的数据特征是什么?
- 应用: 推荐系统是整个应用的哪一部分?

基于上述的三个维度, 我们提出了创建外部环境的模型, 并且在这些模型的基础上设计推荐系统。

本章接下来的内容将会详细阐述这一过程。在 10.3 节, 首先描述了在开始推荐系统设计之前, 应当首先建立的三个模型, 并说明它们对推荐系统设计决策的影响。在 10.4 节, 说明这些模型如何有助于推荐系统的评估。最后 10.5 节提供了对这种方法的一个实际应用案例。

### 10.3 理解推荐系统的环境

正如在前面内容提及的, 我们提出定义三个模型(用户、数据和应用)。这些模型将会在决策制定过程中协助推荐系统的设计者, 帮助他们理解所设计的系统的主要限制条件, 帮助他们在以下三方面提出正确的问题, 界定主要的限制条件: 推荐算法方面的选择, 推荐系统的架构方面的选择以及用户信息方面的选择。我们用来定义环境模型的方法包括定义每个模型的关键内容和设计过程中提出的主要问题。

#### 10.3.1 应用模型

尽管推荐系统自身就是一个复杂的软件, 其本质是一个更大系统的一部分。推荐系统

仅是整个应用的特征之一。它可能是一个次要功能，也可能是一个主要的卖点；系统应用可能是在推荐系统之前就已经存在的，也可能是与推荐系统同期建设的，但无论如何，推荐系统设计必须整合到主体应用设计之中。本节沿着两个主要脉络研究主体应用对推荐系统设计的影响：推荐系统的在主体应用中的角色和主体应用的实现所带来的影响。

表 10.1 应用模型

模型的特征	可能的值
推荐目的	主要业务，关注于“长尾商品”的，增加收入，增加用户黏性，提高系统效率
推荐类型	单物品、多物品、列表
是否具备导航特征	是、否
性能指标(正确性、透明度、惊喜度、风险大小、反应速度、应对攻击的健壮性)	每项性能目标都有其对应的指标
设备对应用的支持度	固定的、移动的、混合的
用户数	单个、群体
应用的基础架构	基于浏览的应用、分布式应用
屏幕空间	有限制的、无限制的

### 10.3.1.1 理解推荐系统在应用中的角色

推荐系统在整个应用中的目标必须非常清晰，这是推荐系统设计之初需解决的首要问题。界定推荐系统的目标通常没有看起来的那么简单，并对所搭建推荐系统类型具有决定性的影响。需要研究两方面的视角：系统应用视角和用户视角。它们有交集，但又彼此独立。用户视角相关的研究详见 10.3.2 节。本节关注系统应用视角。

**推荐目的：**从应用的角度来说，推荐系统可能有不同目的，如：

- 作为应用的主要服务。这类推荐系统在不同领域都有应用，如音乐网站(Pandora<sup>Ⓐ</sup>、last.fm<sup>Ⓑ</sup>、MyStrands<sup>Ⓒ</sup>)和电影网站(MovieLens<sup>Ⓓ</sup>、Netflix<sup>Ⓔ</sup>)。
- 营销“长尾”商品，正如 Chris Anderson 最先提出的那样[5]。其主要观点认为，推荐系统大大降低了用户找到所需商品的难度，对某些电商网站而言，这就是其商务模式的核心。在这种情况下，推荐系统必须关注于那些很少有人问津的物品，将其准确匹配给每个用户。
- 提高用户忠诚度：那些最贴近用户需求的服务会引来“回头客”。用户参与推荐过程可以提高用户忠诚度(如要求用户评分、用户手工填写的信息、突出新的推荐结果等)。
- 通过提升特定产品的销量增加收入。在这种情况下，推荐系统同时由用户偏好和特定营销目标的市场规则所决定。为了保证系统的用户体验，必须小心谨慎地权衡用户期望和商务策略。
- 提高系统效率。由于推荐系统有助于用户更直接地找到所需内容，推荐系统可降低数据传输量，从而降低系统运行成本。

Ⓐ Pandora Internet Radio: <http://www.pandora.com>

Ⓑ Last FM: <http://www.last.fm>

Ⓒ MyStrands, 社交推荐及发现: <http://www.mystrands.com/>

Ⓓ MovieLens, 电影推荐: <http://www.movielens.org/>

Ⓔ Netflix: <http://www.netflix.com/>

**推荐类型：**一个推荐系统可以提供多种推荐类型，从推荐单个物品(或者简单的物品列表)到一个推荐序列(例如，旅行推荐系统会推荐一个旅游地点序列)。单个物品或者列表推荐不用考虑用户在某一时刻的选择是否会影响后续一系列选择。推荐类型抉择是由需求或推荐过程中既有的逻辑规则驱动的。例如，在一个旅行推荐系统中，一次旅行可以被看成一系列出游步骤(如参观博物馆、去海滩等)，这些步骤可以通过像地理、文化、历史、休闲等多种逻辑特征组合起来，以构成好的旅行体验。当用户刚进入一个新的领域，并且需要在多种物品中选择一条最佳路径以实现其个人发展目标时，推荐序列可能会相当有用：推荐结果的逻辑次序是按照用户的学习曲线展开的，即按照用户最容易理解的方式提供最适合的下个行动步骤。符合逻辑的推荐结果通过提供下一步最适合的步骤来帮助他们完成自己的学习路线。遗憾的是，至今推荐序列方面的研究成果很少。我们正在就此进行一些研究；也可能会用到一些来自数据挖掘领域的技术，如 Apriori 关联规则算法[2]。

**整合内容导航功能：**另一个需要研究的重点是，推荐结果如何与其他内容导航功能进行整合。大多数情况下，除了获得推荐结果，用户还有多种方式浏览内容。内容导航功能有效整合能够极大地改善用户体验。

- 用户可能会将请求推荐和浏览内容两种行为完全分开。如果推荐是应用的核心功能，这种情况可能是很不错的选择。这类推荐功能通常出现在网站首页或者应用首屏中。
- 根据当前交互的上下文进行的推荐也是大有裨益的。典型情况是推荐与用户当前浏览内容相似的物品。此时，推荐系统提供的推荐结果能够匹配当前交互的上下文，如当前浏览音乐类型。
- 另外一个要考虑的重点是推荐系统在当前交互模式中属于可选的还是强制的。因为这会严重影响系统的预期可靠性：如果推荐系统成为网站中完成某种任务的唯一方式，一旦推荐结果不准确，推荐系统就成为导致用户满意度降低的主要原因。然而在推荐系统设计中，推荐系统通常与传统的多种站内导航方式同时存在，推荐系统对整个系统的负面影响可就大大降低了。

**性能标准：**一旦明确了上述目标，就可以依据大量的指标来定义系统的性能目标。这些指标不但可以用来评估推荐系统，也是选择合适算法的关键。有很多指标可以使用，文献[25]提供了对许多可行的指标的一个全面的参考。一些关键的指标包括：

- **正确性指标(correctness metrics)：**如精确度(Accuracy)、准确率(Precision)和召回率(Recall)：这些是用来评估推荐算法的技术性指标，一直是推荐系统研究的关注重点。然而，这些指标实际上不足以评估用户满意度[33]。
- **透明度和可解释性(transparency and explainability)：**对于推荐系统而言，让用户理解推荐结果是如何产生的到底有多重要？对某些算法来说，实现高透明度是尤其困难的。例如，协同过滤算法本质上是很难透明的，文献[24]只提供了对问题的分析和部分解决方案。
- **惊喜度(serendipity)：**用户会不会对一些推荐结果感到喜出望外，或者哪些显而易见的推荐是用户想要的？经典的协同过滤算法倾向于推荐那些非常热门，可能被认为是非常明显的，对大多数用户没什么用处的物品。现存技术已经能够纠正这一倾向，详见文献[25]。
- **风险控制(risk taking)：**与前述指标有关，推荐系统是否只能推荐出那些与用户偏好高度相关的物品？如果推荐的目的是帮助用户发现那些在没有系统帮助的情况下用户不可能知道的内容，那么更高风险的物品也是可以推荐的。



- **响应速度/性能(response speed/performance)**: 在很多时候, 应用的反应能力是主要的关注点, 有时甚至比推荐结果的准确度更重要。了解推荐系统要使用的单位时间, 有助于更好地选择算法或决定推荐结果是否应该预先计算。
- **可靠性(reliability)**: 在既定应用环境中, 推荐系统的输出关键是什么? 例如, 一个电子商务网站的推荐系统设计肯定不适用于医疗系统中的器官捐赠配型系统。
- **应对攻击的健壮/健壮性(robustness to attack)**: 尤其是, 具有商业色彩(如推荐产品的购买)的推荐系统, 为影响推荐结果, 它很有可能受到攻击。文献[34]对推荐系统中有可能的攻击进行了详尽的分析, 并提供了针对协同过滤算法的若干解决办法。

### 10.3.1.2 理解应用实现的影响

除了用户可见的特征, 应用实现过程中的一些方面也对推荐系的设计有很大影响。

**单一或多种终端设备**: 同一应用可能通过单一或者多种终端设备进行访问(例如, 一个新闻推荐系统可支持手机、个人电脑、机顶盒)。必须考虑推荐结果是否是基于用户语境的(参见 10.3.2 节)。但就实现过程而言, 随之而来的一些问题是: 用户偏好信息是与基于语境的推荐融合呢, 还是作为独立因素支撑基于语境的推荐? 用户偏好信息存储在何处? 如果用户偏好信息存储在服务器中, 那么向服务器传输是实时的(意味着持续调用)还是批量更新? 即使应用程序设计之初不支持多种设备, 这些问题还是相当重要的, 因为从 Web 应用程序衍生出手机版本已经变得顺理成章。

**单个或多个用户**: 相反, 同一种终端设备可能被多个用户使用。10.3.2 节已经讨论过用户社会环境的影响。此外, 与个性化应用进行交互的用户可能是注册用户也可能是匿名用户; 用户的交互频率是偶尔或者经常。这方面的因素都将影响推荐系统的架构(不同的交互情况意味着不同的识别用户方法, 例如, 是根据登录还是通过 cookie 识别)、算法选择(例如, 对匿名的临时访客采用对话(session)识别的方法, 对注册用户使用用户信息)以及算法参数(例如, 系统对用户的适应程度——也就是用户信息的适配速度: 短期或者长期, 取决于用户访问频率)。

**应用程序基础架构**: 应用程序的基础架构限定了推荐系统可使用的算法类型和推荐系统的一些特殊实现方式。尤其是, 在可扩展性方面需要慎重考虑。两种可识别的主要场景是: 基于用户浏览的应用和通过终端本地程序的应用。

- **基于浏览的应用(browser-based application)**。在基于浏览应用的情况下, 客户端的进程几可忽略不计。单点可调用全部现有信息, 可以使用任何算法。然而, 系统的扩展性是设计时的主要关注点。
- **分布式应用(distributed application)**。当用户终端通过本地程序接入推荐系统时, 可能会使用不同的架构。为了找出最适合的分布式架构, 需要考虑以下准则:
  - **相关设备的处理能力**。客户端设备能否支持密集型任务? 服务端的计算资源能否随用户数增长按需扩展?
  - **网络连通性**。网络连通是否是永久的? 用户要为数据传输付出代价吗? 对移动设备而言, 网络连接对电池续航能力的影响是什么?
  - **数据源**。如何访问提取出来的推荐结果? 是从数据库(信息检索)还是从数据流(信息过滤)中提取?

内容过滤算法可完全运行在终端设备上。这样做的好处有: 因为无须传输用户个人信息, 隐私得到了保护; 架构的扩展性非常好, 因为用户数对服务器负载的影响很小。另一

方面，像协同过滤这样的算法则需要校验全量用户或大量用户的数据。在这种情况下，一个完全的集中式架构可能更合适，和基于浏览器的应用具有一样的优缺点，只是需要特别考虑基于客户端应用的数据更新机制。更复杂的架构也是可行的，如 TiVo 电视节目推荐系统[3]。在这个系统中，服务端进行部分计算，其余的计算是在每个机顶盒(客户端)完成的。其他的复杂架构包括分布式配置管理[13]，或确保在不同的终端上运行的推荐系统进行会话的机制[28]。基础架构的选择也可能受到与其相连的系统其他组件的影响，如外部网络服务。

**屏幕空间：**设计初期很容易被忽略的一点就是推荐系统将用于多大的屏幕空间。大多数情况下，屏幕空间是非常有限的，并且受到应用的 UI 设计的限制。这不仅是一个量化议题，并且对所提供的推荐结果的本质有影响。例如，假设推荐系统试图提供偏向于探索性的推荐结果，那么必然要有足够的屏幕空间以展示足够多的推荐结果。在向用户展示推荐描述的环节中也面临同样的问题，文献[40]提供了一种在小型设备上展示用户配置的解决方案。

研究推荐系统的应用仅是了解推荐系统设计的一系列限制条件的开始，只了解这些还不足以建立一个合适的推荐系统，还需要掌握更多关于用户和数据的知识。

10.3.2 用户建模

充分理解用户是任何推荐系统成功的基石。对最终用户进行深入研究以便用户建模的工作在开发周期中应该早点开始。用户建模是推荐系统设计决策过程中，影响技术选择的关键因素。在任何项目的初始阶段就采纳以用户为中心的方法，都能极大减少有关大规模的重构、系统维护和用户支持的需求[9, 22]。

本节提出通过那些可能对推荐系统的设计和开发者所面临的选择有影响的特征来刻画用户(表 10.2)。

开发推荐系统时，用户理解的基本维度是以“人为中心”的方式[26]，清晰地理解与尽可能详尽定义用户应用(推荐系统所在的)交互系统时，其使用环境是什么？最终用户是谁？促使用户使用该系统的推荐服务背后的用户期望和目标是什么？系统应用的核心环境因素

是什么？在全面回答有关使用环境的问题过程中，有关系统设计和选择的基本要求就会被揭示出来。

10.3.2.1 理解谁是用戶

理解谁是推荐系统的用户需要仔细斟酌三个主要方面：理解识别用户的主要特征，用户的技能水平和用户之前在类似系统的使用体验。我们强调用户特征的识别，是因为其在推荐系统设计中具有独特用途。

**识别用户特征：**在推荐系统的用户模型开发中，首要起点是为不同的用户群画像，包括采集用户人口统计学信息(如年龄、性别、工作领域、国籍、语言)以及来自用户调研的深入定性信息。理解这些因素能够让开发团队建立用户之间的联系，以及鉴定用户的需求。

表 10.2 用户模型

模型特征	可能的值
人口统计信息	是，否
是否有目标	是，否
目标性质	隐性，显性
期待水平	高，中，低
处理用户期待是否需随时间变化调整	是，否
用户终端条件限制	是，否
用户情境重要性	高，中，低
社交环境	独自，与他人一起
信任和隐私关注度	高，中，低

用户聚类能够：1)建立基于人口统计信息的简单推荐系统。将用户聚类为不同群组的方法通常用于目标广告投放解决方案[23]；2)定义一个用户原型(stereotypes)[42]：原型技术使用一系列不同特征定义一个用户群，当系统的新用户出现时，可根据新用户的个人数据，预判其用户类型，推荐系统按照其用户类型分配给他一系列默认的偏好值，而推荐系统的用户信息更新机制将依据用户随后产生的行为对其用户信息进行修正[17]。基于用户特征的个性化解决方案可与更复杂的技术组合使用：如作为混合过滤程序中的初始步骤，或通过原型用户信息来引导基于内容的过滤算法。

此外，用户类型(如专业用户或最终用户)是判断用户期望水平等方面及相应算法选择的关键指标。

### 10.3.2.2 理解用户动机、目标和期望

**目标和动机：**推荐系统的设计者需要识别用户任务[25]和了解应用程序能否完全支持。例如，亚马逊网站就提供用户可能购买的产品并且推荐他们去购买。从用户的观点来看，他们使用这项服务的初衷无非有两种情况，要么为自己买，要么为他人买。然而亚马逊的推荐系统没有区分这两个目标，因而导致在其中一种情况下提供的推荐结果不够准确。另外一个例子，带有内容推荐系统的搜索引擎，能够提高用户浏览因特网并且根据其搜索请求发现信息的机会。在此背景下用户的动机可能是需要完成一个有针对性的目标，或者他们的目的仅仅是放松、花时间浏览些有趣的事。识别和理解用户动机可以使推荐效果和用户体验得到提升。通过用户模型(表 10.2)来深入了解用户，可以让设计者考虑这个应用将来需要的支持可能的任务范围。

在大多数情况下，推荐系统的使用可能会有很多目的，而设计者必须考虑发现本质目标的方法，明确的还是隐含的。一个明确的目标可能被一个应用定义(如亚马逊可以添加一个按钮来询问用户是为自己还是为他人购买这个物品)或者让用户来表达(如通过一系列的问卷)。一个隐含的目的可以是应用本身事先定义的(例如，在个性化的新闻推送系统中，系统提供新闻，认为用户想要得到更多世界大事的详细信息)或者从与用户的交互中来推断。相反，得到一个准确的任务目标可能不太容易(例如，应用可能被兴趣所决定)。在这种情况下，建立一个好的推荐系统是很困难的，因为用户的满意度和外界的因素息息相关，如不受系统控制的用户心情。在这种情况下，多样性、数量和精细度可能是推荐系统最重要的属性。

用户目标对于在展示推荐结果时的过滤算法和多样性试探[55, 54]非常重要，如果在设计阶段没能很好地解决，就会产生用户不满。例如，基于内容的模型更适应集中的人群(因为集中性的结果)，而协作模型更适合那些没有集中的目标的人群(因为更广泛的多样性的结果)。

**用户期望：**在前面的内容中描述的用户显性或者隐性目标是评估用户期望等级的关键：

- **高期望值**可以被理解为那些目标很明确的、专注于完成当前任务的用户。这意味着推荐系统必须提供高质量[25]的推荐结果才能得到用户的高满意度。这种水平的期望值也与应用层的决策有关(见 10.3.1 节)，推荐系统在整个应用中的地位以及目标设备的约束对用户期望都是极其重要的。回到新闻推送系统的这个例子上，也就是说这个系统必须提供全部是好的项目列表。如果用户在推荐系统给出的前十个推荐中找不到合适的个性化新闻，最好的可能是他们开始向下翻动继续寻找，最坏的可能是不再使用这个应用，因为他们在第一次使用时就觉得不满意。

- **中等期望值**可以看作推荐系统推荐一些好的物品[25]。如果用户有选择并且在使用推荐系统中灵活地完成了任务,那他们的期望值会相对低一点。在这种情况下,我们也会发现,使用推荐纯粹是评估系统给予的推荐结果是否跟他们的期望一致。一些人在寻找与他们偏好完全一致的推荐;另一些人想发现与他们目前习惯不一样的新建议。
- **低期望值**主要是针对那些被用在一个随机的环境中的个性化的应用。正如[46]对网站推荐系统的描述中所说:“研究表明,如果任务是固定的,如仅是随意地看看,那么推荐网站是困难的。”

每一个不同等级的期望值导致了不同的态度,从最终的用户驱动、用户的满意程度,删除应用到使用推荐系统仅是为了娱乐。

**处理随着时间改变的期望值:**当使用推荐系统的时候,用户的期望可能随着时间的改变而改变。考虑是否需要建立一个自动适应的推荐系统是相当重要的。大部分的推荐系统已经包含时间变量;随着得到用户信息的增长能够得到更准确的推荐结果。第一次使用推荐系统时,用户在一开始使用推荐系统时对推荐系统的能力也抱有期望,随着时间的增长,他们对推荐系统更加熟悉,他们的需求也发生了变化。不但推荐系统能够随着时间的变化来满足用户的需求是非常重要的,而且他们能够在用户的生命周期中尽早地证明系统的性能来满足用户的期望,建立对推荐系统输出结果的信任和确保继续使用也是相当重要的。

### 10.3.2.3 理解用户环境

最后一个要考虑的地方是推荐系统使用中的环境因素:

**用户设备:**首先要考虑的是用户用什么设备来接入推荐系统?现在在应用和服务器上的推荐系统有了新的特征就是方便接入,如手机、电脑或者机顶盒。这些系统设计的实现已经在10.3.1节学习过。

**交互环境:**环境的考虑首先包括位置,因为无处不在的移动设备的使用使得它变成首先要考虑的问题。位置的考虑因素可能包括绝对地标,例如,如果你希望在移动设备上推荐本地的名胜古迹,并且在绝对的位置理解用户的行为,如了解用户是在工作还是在购物?另一个重要的环境因素要考虑的是时间因素。给用户建立模型的时候,对于推荐系统考虑收集带有时间信息的用户属性数据集是重要的。举个例子,一个用户可能在上班途中通过移动电视看新闻,但他更喜欢在同一趟火车的下班路上看喜剧。已经有一些解决方案在推荐系统模型中应该考虑这些因素;例如,兴趣和给定的环境有一个显性的联系[12],或者属性和环境之间的联系较为隐蔽[29]。结合位置和时间的数据集对于建立更加健壮的推荐系统具有极大的帮助[4]。

**社会环境:**这里有一个要考虑的重要因素是推荐系统的使用是单个人还是和其他人一起。这影响许多的设计决策:包括算法的选择、数据收集的方法和推荐结果的展现。举例来说,这些问题可以有助于决策出成组推荐的策略(也可以阅读第21章),例如,合并用户单独的偏好来获得一个群体用户信息,可以用于内容检索进程[6, 10, 20],或者为了通过用户合作定义一个共享的内容检索策略[7, 32]或者其他描述方法的一致性原理应用[15]。然而尽管在一个群体中,人们仍然需要独立的推荐结果,例如,在电视节目[8]。对于这个例子,在[11, 30]已经提到了一些解决方案。

所以这些问题需要经过充分的考虑,并且尽可能地做大量的用户研究。尽管这看起来是一项非常大的动作,但是理解这些领域中哪些是与推荐系统相关的,可以提炼大量的研

究需求。

### 10.3.3 数据模型

设计者应该仔细研究的最后一点是推荐系统将要利用和管理的物品特性。事实上，对物品的描述比个性化推荐系统存在要早，并且推荐系统的设计者也不太可能影响或者改变它们。我们提出了一个数据模型用于定义那些会对系统设计和将来的推荐结果有影响的主要数据特征。设计者应该遵循我们的数据模型，比如，对于数据模型中的每一个特征，都应该考虑其在表 10.3 中可能的值。

#### 10.3.3.1 理解可以用来描述物品的数据类型

这里存在以下几种方式来描述物品：

- **无结构化数据：**物品只能用无结构化的数据来表示。无结构化数据指的是那些既没有一个数据模型，也不能很容易地被计算机程序使用（如视频、音频、无结构文本）的数据。在这些情况下，需要通过预处理来提取标志性的关键词或概念以帮助区分不同的物品。事实上，无结构化数据并不是阻塞所在，因为在大多数情况下，有许多的技术用来获取描述物品的元数据集。在文本分析中，像 GATE[19]或 Lucene<sup>⊖</sup>等工具能够从无结构化的文本中提取关键词。虽然从其他类型的数据如多媒体内容（图像、视频）中提取元数据的技术缺乏可靠性，但还是经常需要将它们结合在一起。
- **半结构化数据：**一个物品经常用好几类元数据描述，对应的主要物品特征项和自由文本。半结构化数据的典型案例是亚马逊<sup>⊗</sup>的产品页或者被很多属性（节目类型、节目时间表等）所表示的电视节目。每个属性取值在一组有限的词汇（这属于一个分类或本体），或者包括无结构元素（如节目的大纲）。在这种情况下，数据数量和质量的评价需要知道物品的元数据是否应该在非结构化部分通过执行一些处理和分析来提升。
- **结构化数据：**物品也可以用结构化的模型所描述。这可能是一个标准或者一个事实上的标准，例如，TVAnytime<sup>⊗</sup>用电子节目指南（EPG）来为 DVB-H 描述电视节目。当有结构化数据的时候，设计者必须检查数据的数量和质量来预计在一个个性化系统框架信息中缺失的可能性。

这个基本的数据特征有许多影响。首先，在算法方面：如果数据是无结构化的，那么提取相关元数据的开销会很大，效果也不太准确。下面的全部推荐算法系列应该排除：基于内容的推荐系统算法[51]（第 3 章），贝叶斯模型[21]，基于规则的系统 and 度量距离。只有协同过滤模型（第 4 章）能够用在这样环境中。第二，在用户建模方面：用户模型中应该

表 10.3 数据模型

模型的特征	可能的值
数据类型	结构化、半结构化、无结构化
元数据的质量和数量	高、中、低
元数据的表达方式	基于关键词，基于语义
基于标准的描述	是、否
物品的体积	大、小
物品的种类	同类、异类
物品的分布	长尾、主流
物品的稳定性和持久性	稳定的、变化的、易变的
用户评分	隐性的、显性的、无
评分类型	二进制、多等级……

⊖ Lucene，一个开源的信息检索库，<http://lucene.apache.org/>

⊗ Amazon，<http://www.amazon.com>

⊗ TV Anytime，<http://www.tv-anytime.org/>



反映数据表示方法,例如,如果一段文本表示为一个关键词向量,那么基于方法的内容将使用关键词向量来表示用户信息。在无结构化数据情况下,用户模型可以非常简单,如用户的消费历史。

### 10.3.3.2 理解元数据的质量/数量

结构化数据的质量和数量对于推荐系统是一个重要的性能因素,尤其是在使用基于内容的推荐系统时。元数据的某些属性在关于个性化系统设计的选择中扮演着非常重要的角色。

**质量:**设计者需要对可利用的元数据的质量有一些掌握,了解做决策时要考虑的元数据的质量是中等的还是低等的。通常情况下,如果一个物品的元数据描述能够将这个物品与其他物品区别开,则认为物品的元数据描述是高质量的。例如,在新闻 RSS 提要中,两个物品能分享共同的关键词,相当于新闻的一类(如运动新闻、足球),但是这必须有足够的其他关键词来将新闻区分为是一些比赛相关的,是全球性的竞赛事件相关的,还是一名足球运动员相关的等。在这个示例中,设计者必须理解在描述中细节的正确等级来获得推送新闻之间的不同。在我们模型中,对于推荐系统的设计者而言,元数据的质量特征是决策的一个主要点。设计者不得不在推荐结果的准确度与推荐系统响应时间、存储能力和进程开销之间进行权衡。例如,如果设计者想要一个最佳的性能,他将在架构上引入一些限制,避免在轻量级客户机上运行。设计者也可以选择用两阶段式的算法运作推荐系统,分布在服务器和客户端设备之间[37]。质量特征也和推荐系统在实际应用中所扮演的角色以及用户对于这个角色(请看 10.3.1 节和 10.3.2 节)的期望有关。

**表现力:**未来的用户也是评估元数据进程的基础。元数据必须能够反映用户对物品的观点,并且表达出用户对物品不同特性的想法。所以元数据的**表现力**对推荐系统的性能是至关重要。物品可以用大量的语义表示:从没有语义使用标签/关键词,如 Flickr<sup>⊖</sup>上图片的注释,到更为高级的知识表示,如分类或者本体论(如 OpenCalais<sup>⊖</sup>中的文本分类)。作为一个经典案例,比较基于关键词的方法和语义方法,一个物品被一些特殊的关键词(如足球、排球、篮球等)引用,那么这个物品不会被推荐给对更高层次(如团体运动)感兴趣的用戶。通过语义表示,像足球、排球、篮球隐含知识表示为团队运动,可以被用来提取用户潜在的偏好和推荐结果,因为这些运动之间的语义关联性,所有团队运动相关的内容都会被推荐。通过用语义概念来描述的元数据能够用于更为复杂的推荐算法,如文献[47],还要根据位置考虑在同一层次的相关类似物品的存在;或者文献[48]所描述的延伸语义选择。对于协同过滤技术,在算法流程中考虑语义的相似性,已经证明可以提高准确度,能够帮助减少稀疏度的问题[35]。然而,这里也有些实际性的因素要考虑:例如,语义的推测引入了一些处理开销,可能不太适合小型设备。另外,如果元数据没有被语义很充分地描述,一些算法能够提高元数据的表达水平,例如,通过维基百科[18]<sup>⊗</sup>将本体内容与社会化标签相匹配,或者创建大众分类(相比较第 19 章)。

**数量:**元数据的数量是要考虑的一个很重要的因素:太少的元数据可能导致推荐结果的不准确,而太多的元数据将导致无用的处理。另外,元数据对物品深度(精确度)和宽度(不同的描述)的描述可能会变化。这样描述的风险是推荐的物品可能与用户的期望并不一

⊖ Flickr, <http://www.flickr.com>

⊖ OpenCalais, <http://opencalais.com>

⊗ Wikipedia, <http://www.wikipedia.org>

致。例如，如果一个用户对有关自然灾害的新闻感兴趣：对于深入的描述内容，用户将得到的推荐结果是关于不同种类型的灾难；对于广度的描述内容，他将得到对于灾难不同的观点（政客、政治学家、经济学家等）。深层次的描述可能会加强基于内容的过滤算法的某些缺点，事实上也过于专业化。用户想要的深度和广度的层次可以通过系统来学习。

**基于标准的描述：**不考虑元数据的表现力级别，元数据可以用不同的方式来描述：使用标准，如 Dublin Core<sup>⊖</sup>，MPEG-7<sup>⊖</sup>、IPYC<sup>⊖</sup>；或者使用专有格式。如果元数据的表示不受个性化应用的设计者左右，那么可以用某些要素来指导选择元数据的不同形式，如推荐系统与应用中的其他部分的整合程度（如使用一个检索引擎）。

### 10.3.3.3 理解物品集的属性

**物品的数量：**除了每个物品元数据的数量，我们也应该考虑数据集中物品的数量。它是决定选择推荐系统家族的重要因素。实际上，协同过滤算法需要大量的数据集或者用户集来计算相关有效性。这特别适合书籍、电影等的推荐，而基于内容的小型数据集也可以处理（如电视节目）。

**物品的分布：**数据集中的物品分布情况也是一个必须考虑的因素。例如，在一个电影数据集，如果有很大比例的电影被标注有“动作片”的注释，这种元数据可能描述得不充分（如果他喜欢动作电影，那么有太多的电影要选，如果不喜欢，就基本上没什么可选的了）。在这种情况下，需要重新考虑数据注释的深度层次以便获得更高质量的元数据。

**物品的本质和物品集的稳定性：**新的物品，图书，或者电视节目有本质上的不同，因此依据物品属性的不同用户行为也不一样。例如，很容易理解对于电视节目和图书的兴趣比对于新闻的兴趣更加稳固，因为新闻变化得更频繁，并且有更多不同的主题出现。所以，这项准则影响是否要用可适应型推荐系统（例如，用户信息随着时间而改变），重点是需要了解不同的发展模式，如稳固的兴趣、发展的兴趣、快速变化的兴趣等[38, 39]。我们也应该考虑物品集的稳定性，例如，引入新物品或者物品消失的频率。另外，一个推荐系统可以在同构数据下工作（如 MovieLens），或者也可以用异构数据（就像亚马逊那样做的）。为了给物品推荐系统选择一个适合的策略，研究物品之间的相关性是很重要的，例如，如何在书籍的品味中反映对于某种音乐类型的兴趣。这样的分析能够帮助选择推荐系统的策略，如综合考虑所有的物品，或者用基于每一个物品类型的特性应用特定的准则来替代。

**用户物品评分：**另一个要重点考虑的是用户是否有评分。如果在系统中用户没有对相关物品进行评分，这就排除了整个协同过滤模型家族了。用户对物品的评分既可以很显式的（例如，在亚马逊，用户可以指出他们对一本书或者一盘 CD 的喜欢程度），也可以用其他的表示准则（如二进制、多级等）。如果这不能被直接地应用，但也认为在个性化应用是有用的，它可以被用来计算隐式反馈指标[41]（例如，购买物品这一事实可以被认为是在用户兴趣的隐式指示），这个也可以用在一些推荐算法（如协同过滤算法），或者作为用户信息适应模块的一个输入，即根据用户对特定物品是喜欢还是不喜欢来更新用户的兴趣。

### 10.3.4 一个使用环境模型的方法

在表 10.1、表 10.2 和表 10.3 中，我们引入三个模型来帮助理解将来推荐系统的环

⊖ 都柏林核心元数据倡议，<http://dublincore.org/>

⊖ MPEG-7 概览，<http://www.chiariglione.org/mpeg/standards/mpeg-7>

⊖ 国际新闻电讯委员会，IPTC(International Press Telecommunications Council)，<http://iptc.org>

境。对于每个模型和每个特征，我们提出了一些指导方针来定义将来推荐系统的需求和约束。为了充分理解这些特征的重要性，我们分两步提出一个模型。

- **确定不同特征之间的依赖关系：**设计者必须建立一个贯穿三个模型的依赖图来找到哪些特征会影响其他的特征。这个图将帮助设计者理解一个特征的变化是如何影响整个推荐系统环境的。例如，改变整个推荐系统的导航功能(应用模型)将改变用户期望(用户模型)。
- **确定模型的关键特征：**关键特征是指那些对推荐结果的选择、自适应算法，以及推荐系统架构有重要影响的特征。例如，正确性(应用模型)对推荐结果的选择有很重大的影响。这些关键特征的确定有助于准备评估框架的工作和如何解释评估结果。

在 10.5 节会给出一个例子，作为这个方法的例证。

然而，在这一节中，我们已经确定了所有在设计推荐系统时要研究的要素。基于第一步，设计者应该能够为过滤物品选择合适的算法，为用户信息做调整(如果有必要)，以及选择正确的系统架构。下一个阶段设计者要考虑如何实现他的推荐系统(见参考文献部分)，然后评估算法，正如在 10.4 节中所阐释的。

## 10.4 在迭代设计过程中理解推荐验证步骤

尽管提出的模型能够对推荐系统的关键部件做出精明的决策，但是在使用之前，还需要通过实验测试及必要的迭代设计，对许多参数进行调整。以下两个方面必须注意：算法的自我验证和推荐结果验证，这个过程必须让用户参与。

### 10.4.1 算法的验证

许多实验性论文已经证明针对现有数据集推荐方法的测试是有用的，例如，MovieLens 和 Netflix<sup>①</sup>数据集等。从这些实验中，这些数据集被用来理解算法的运行状况；有时候，为了能够使用这些数据集，需要把它们进行转换和填充(例如，使用 IMDB<sup>②</sup>数据填充 MovieLens 数据)。在许多文献中已经对这些实验数据进行了深入的研究。例如，在文献[25]中，详细列出了一些可计算指标。此外，这些指标支持基于数据特征和有用信息其他方法的客观比较，并且当需要大规模的用户时，这可能特别有用。这样的评价指标也可以针对算法中个性化参数进行修正，例如，距离度量可以用来计算一个用户与相关项目内容之间的相似度。这些方法在许多文献中被广泛提及，因此在这里就不再详述。具体细节可以参考第 8 章的内容。

在可用数据集上测试算法的详细过程中，由 10.3.4 节中描述的方法确定特征关联和影响图，必须帮助设计者能够发现数据集是否正在改变相对于目标系统的推荐环境(数据、用户和应用)。这个图必须能够帮助解释说明算法的结果，并且决定算法修正的范围及测试的走向。如果环境特征确实能够接近一个目标系统，那么可能值得尽量调整算法。但是，如果实验环境与目标环境分析结果存在显著的差异，那么这个实验过程可以尽快结束，并把注意力放在故障的排查上，而不要放在小幅度地提高推荐的精确度上。许多的研究都关注提高算法的精确度[33]，但即使一个推荐算法能够提供准确的推荐结果，也仍需要考虑数据实例、用户模型及其与数据集的关联性。因此，理解环境模型特征的重要性及

① 用于竞赛的 Netflix 数据集，<http://www.netflixprize.com/download>

② IMDB, the Internet Movie Database, <http://www.imdb.com/>

其相互关联性对评价结果的正确解释要比一个标准的数据集更重要。例如, 当我们把电视数据转换成新闻数据时, 用户的行为是不一样的, 并且数据也不具有相同的动力学特性(具体见 10.5 节)。在这个例子中, 在调整算法提高其在电视数据集上的准确率中花费大量的时间完全就是浪费。

### 10.4.2 推荐结果的验证

有许多技术措施能够在理论上提高推荐的准确度。为了改善性能和提高用户满意率, 对性能和满意率的保障措施严峻考验在于, 能否切实体现在所有领域推荐系统的评估结果上。

评估任何交互系统的关键在于服从一个以用户为中心的方法。也就是基于变化和改进行进行尽早的、反复的、频繁的评估。这带来一些挑战。事实上, 正如前面所讨论的, 推荐系统很少是开发活动的中心, 而是一个更大的产品或者系统集成的子功能。常见的情境是开发更大的主机系统在不同开发时段运行推荐引擎。这种情况使得对于潜在推荐技术的评估, 没有方法收集建立用户信息或者向用户呈现推荐方法的用户偏好。

这些因素往往会导致面向用户交互的推荐(数据收集和推荐结果展现)评价与底层引擎分离, 除非这两个过程能够实现原型集成。从评估的角度来说, 这是有意义的。例如, 能够防止研究成果与其他领域具有竞争性的积极或者消极设计产生相混淆。

在用户评价工具包中有许多技术可以用来研究推荐系统的发展状况。下面将介绍一些有用的可选择方法及其最好应用时机和上下文背景。所有这些方法都已经被应用到实际项目中, 并且在推荐系统领域中基于我们指导用户的研究经验。

#### 10.4.2.1 卡片分类法

卡片分类法不是一种严格意义上的评估方法, 但是在实际中是一种有用的形式设计研究工具。它可以用于根据用户与内容或者概念关系模型对项目进行分类。此方法基本是要求用户依据相似度对一组卡片进行分类, 并描述项目内容或者子类[44]。为了能够在分类用户发现共享模型并建立树型关系, 可以利用聚类发现产生一组用户的结果排序分组。可以使用像 IBM 的 EZsort 这样的软件工具来大幅度地降低此分析过程的时间和复杂度[20]。

当我们将应用于电视类型分类的距离函数进行研究时, 就已经用到了此方法。尽管此领域中存在多种元数据结构, 如果测试不能与用户对任一类别间相似度的理解相结合, 那么这种测试是相当困难的。例如, 研究结果表明由于用户要表示更轻松的表演这个概念, 常常把一些不同类别的物品分类在一起。这些类别包括娱乐节目、喜剧和肥皂剧。相反, 还会把如新闻和记录等真实内容的类别分成不同的类别。此信息将被集成到算法的开发过程中用来提高推荐的准确率。

开展这项工作的价值在于内容物品相似度的度量基于真实用户的分类方法, 而不是基于程序员或者工程师创建的智能结构。此方法还挖掘出不同类别间的主观相对距离。从用户的角度来讲, 这两个因素可以提高推荐结果被认为是相关的和准确的。由于这种方法有益于底层算法的构建, 因此这个工具最好在开发形成过程中使用。

#### 10.4.2.2 低保真原型

低保真原型是指包括与用户探究交互式设计理念在内的一系列方法的涵盖性术语。通常这种方法采用纸上的原型来评估如何给出推荐结果的设计理念, 以及在推荐系统开发的



早期如何获取并管理用户信息。这种方法的优点在于可以使得评价在早期得以实现,不需要在开发过程中耗费太多的时间和代价。但是此方法在评价实际推荐引擎方面没有任何价值。此方法把原型作为一种虚拟提示,以一种半结构化的方式运行。在文献[50]中介绍了此方法及一些有效实例。使用低保真原型关键在于他们不应该看起来像成品设计,因为这在讨论设计思路时会限制用户的作用。简单示意图和轮廓可以使用户专注于有意义的方法改进和精确的设计思路。此方法是从用户的角度试图收集丰富的数据信息,因此从本质上讲它是定性方法。利用一个确定的用户群体中的三四个用户评估一个设计往往会在积极的方面提供足够的洞察力来显著地影响设计。

我们已经利用这种结合用户的研究方法去评估了实际的概念设计和一般的规则设计。例如,为了收集用于构建用户模型的反馈信息,我们利用这种方法说明了设计评分尺度的可能思路。简单的线条图足以让用户分辨各个概念之间的差异,并对每一个的优点和缺点有很好的认识。这可以让用户对为推荐系统提供的显式反馈的概念有更加普遍的思考,并知道这在结果和隐私方面意味着什么。

#### 10.4.2.3 主观定性评估

此方法是从目标用户的角度评估推荐引擎的准确率和满意率水平。它的主要优点在于多功能性:它是一种定性的方法,即使在支持数据集和推荐结果展示部件还不存在的情况下,也能够为一个系统立刻输出推荐结果。它同样可以用于评估充分完全实现了的系统。这个方法是基于主观评价的,应用此方法的一种有效方式是通过主持管理或者参与完成标准的调查问卷<sup>①</sup>。在问卷调查中为用户生成的每一项推荐,并根据用户对推荐的意见提出一些问题。此方法包括三个步骤。

1) **用户数据的收集**,建立用户信息:收集一定数量的数据是推荐系统一个基本性能特征。然而,在从不同的数量的信息中建立参与样本是明智的,这种信息在给定的时间点被获取来代表不同类型的用例模式,例如,两周之后。这需要分析有关用户感知的准确率和满意率的推荐学习曲线的效果。从用户那里轻松地数据收集过程同样也是一个完善的推荐系统的基本功能。如果系统在设计阶段的数据收集功能就是可用的,那么在研究过程中就可以考虑利用它们。由于利用其他方法很难把此类数据从用户那里直接提取出来,所以在系统中只能收集隐式应用数据。显然,问题的关键在于开发团队能够接触到实际用户。因此从直接评估的角度来讲,最初收集的用户数据(例如,web分析形式)是没有用的。从零开始收集隐式数据需要参与者的长期监督实验。这就要求使用像视频观察或者远程使用跟踪这样的技术,从中提取准确的偏好数据。这是一个费力的研究活动,而且也会提供获取真实用户观察的额外收益以及新颖的使用案例,这反过来会驱动创新。相比之下,显式数据收集在没有工作系统条件下也可以较容易地仿真出来。通过电子或纸质的调查表,收集反馈信息是一种有效方法。电子制表软件是应用中的一个简单实例。用户被要求为成品系统中的四十条内容进行显式评分。通过互联网远程招募50个参与者,并用E-mail发送调查。用户填写他们的回答,并将其返回给设计团队。

2) **产生推荐**:从参与者返回的调查问卷中提取用户偏好数据,并将其编码输入推荐引擎中。提供个性化的推荐结果的一个重要考虑因素是跟踪整个研究过程中用户拥有的数据和他们的反馈回应。首先,在研究过程中需要获取保留用户信息的许可认证。给每

① 问卷调查开发是一门科学,所以如果你不是很熟悉问卷调查设计和意见评价的技巧,文献[36]是一份很好的实际参考。



个参与者一个 ID，在这个评估过程中利用每一个文件中的数据。在研究过程中，安全保存参与者的所有具体信息，以防泄露参与者的 ID 信息。最后，在研究结束后，要确保销毁参与者的个人敏感信息，并将对研究结果进行匿名处理。在这个简单过程中保护用户隐私。

3) **推荐结果展现**：一旦为每一个用户都产生推荐，那么这些推荐结果需要返还给参与者进行评估。在调查问卷过程中需要参与者回答的问题取决于推荐系统的目标。关于推荐结果，用户有如下三个普遍的兴趣点：

- 推荐结果是否符合他们的偏好？
- 是否是兴趣推荐？
- 是否对产生这种推荐的系统满意？

对特殊推荐项目的重要问题需要用语言着重标出，而不是把整个推荐列表作为一个整体，因为整个推荐列表不可能对提高推荐系统有所帮助。搜集有关个体项目的信息能够识别推荐设计的弱点和缺陷。例如，有关 TV 推荐的一个项目，就可以研究为什么一个成功的推荐系统能够接受少量的异常奇异满意评分。奇异点研究能够发现不能被元数据确定的外语内容推荐。因此设计团队通过修改，使得系统能够更智能地识别和处理外语内容类别，并解决不满意的问题。

对数据的分析可以解决用户可能的深层次的满意度。这也可以直接比较竞争系统间的结果，这些系统可以应用于基准测试，还可以对设计方向的决策产生反馈。

#### 10.4.2.4 日志研究

一旦一个推荐系统应用于一个领域，从中长远的角度上如何收集信息才能算是一个成功的开发。在系统中收集应用机制不能获取应用系统中的主观动因、欢喜或者沮丧。推荐系统的一个重要考虑因素是在较长时间内对不断变化的用户体验的长期学习。在线系统整体经验的案例分析比简单的满意度更有意义。日记研究可以从一周到几个月做什么事情，取决于研究的系统和参与用户的时机。当直接观察或者监督变得非常困难时，基于日记的研究特别适用于移动上下文环境。我们最近已经使用这种方法做了研究，这个研究涉及在没有远程用户日志的不同移动设备上不同视频内容案例场景的影响评估。此方法被证明在识别用户特定关键点方面有较好的效果。

事实上，日志在社会研究的许多领域中是一种成熟的方法（例如，在文献[10]中的实例）。此方法依赖于用户在与感兴趣系统之间日复一日的互动过程中创建的自我报告记录（它涉及真实用户用其特有的语言描述的上下文环境）。从这些能够被集成提升系统的数据中获得信息，已经为下一代推荐系统的发展用来发现用户需求和上下文问题。根据此方法的流程，以下可能的几点缺陷需要考虑：

- **用户的招募和保持**：在长期研究过程中往往很难招募到用户，而且保留他们更加困难。
- **日记无报告或者虚假记录**：完成整个日志需要许多付出和努力，对于用户来说坚持整个研究过程很困难。

为了成功地完成日记研究，与参与者持久地保持联系是非常重要的。定期面对面或者电话汇报，以灌输研究人员的期望、监督数据的采集、询问最近上报的事件或评论等方式，鼓励用户持续地完成他们的日记。这种策略可以用来区分可能导致用户满意或者不满意情节的关键事件。

同时没有详细的列表，这一系列的方法在开发过程中具有一定的用途，从早期的设计阶段、原型开发到最终发布以及后续的发布版本。

## 10.5 应用实例：一个语义新闻推荐系统

在前面的内容中，我们提出了一种建立推荐系统的方法：通过对三个模型(应用、用户和数据)的实例化，来迫使设计者考虑推荐系统设计可能的选择。在此节中，将从这些模型中获取的东西与一个实际应用系统(新闻专业人士或者最终用户能够创建一个新闻市场，在那里以个性化的方式共享和买卖多媒体内容)内能做的东西进行比较。

### 10.5.1 背景：MESH 工程

通过对 MESH 的描述，说明对推荐系统设计有帮助的模型的有效性。MESH<sup>①</sup>(Multi-media sEmantic Syndication for enHanced news Services)是一个研究项目，在这个项目中我们参与设计了一个用于智能产生和发送语义增强多媒体新闻信息的框架模型[52]。

MESH 的主要目的是为新闻专业人士和最终用户提供有效的方法组织、定位、使用和发布大规模的、异构的多媒体信息。一个 MESH 系统是一个新闻内容传播过程的中间者，能够利用多媒体内容网页为用户提供导航，为用户自动寻找其感兴趣或者需要的信息，并将其以尽可能好的方式展示在任何终端上。核心思想的构成如图 10.2：1)基于半自动语义提取元数据和用户计算偏好，把内容信息分发给用户(推送模型)；2)个性化的多媒体摘要可以轻松承受大规模的信息，通过导航辅助设备，提供进一步使用信息的初始入口连接，避免“网络迷失”综合症；3)通过固定移动渠道，先进的聚合方法可以把新闻从根源(记者)到最终用户快速分发，提出新的灵活协同业务模式。

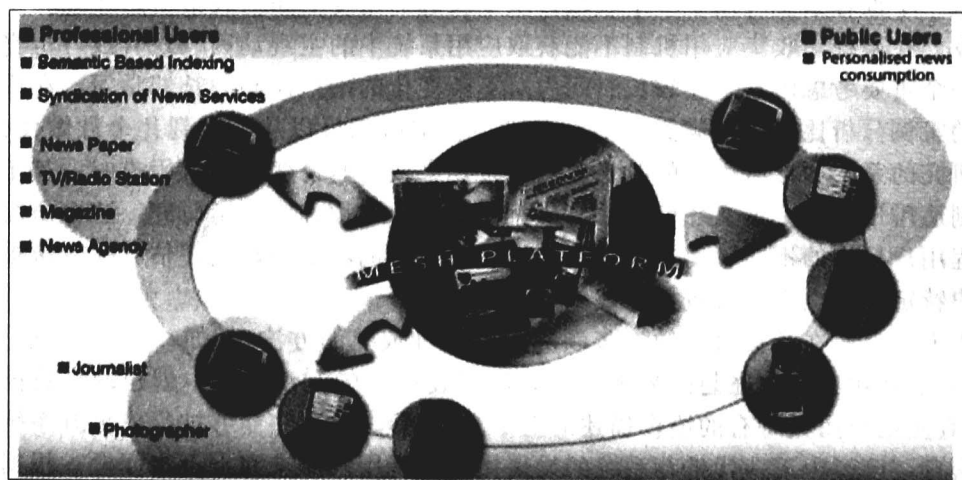


图 10.2 MESH 平台的示意图

在这种背景下，个性化的作用是：1)使系统能够积极推送给用户个性化新闻项目或个性化的多媒体新闻摘要；2)支持用户(专业人士和最终用户)个性化的新闻搜索。

### 10.5.2 MESH 的环境模型

基于上述应用描述，为了确定推荐系统设计的约束条件，我们可以举例说明如何实例化在 10.3 节中描述的环境模型。

① MESH IST project(FP6-027685)(03/2006-02/2009), <http://www.mesh-ip.eu>

## 10.5.2.1 环境模型的实例化

表 10.4 应用模型实例

推荐系统用途	提升系统效率，为专业用户(节约采集新闻信息的时间，如为某个特定的主题建立一个“档案”)，为最终用户(接收或者浏览与他们的资料、最近的兴趣、当前的状态或者最近的查询相关的新闻)
推荐系统类型	多个推荐物品，提供一个多媒体新闻文档的推荐列表
具有导航功能的整合	紧密的，如通过与信息检索的耦合来提供个性化搜索服务
性能准则	取决于系统用户的类型，不同的准则具有一个原始的重要性：记者和专业用户对正确性、响应速度、可靠性和健壮性更加感兴趣，而最终用户对正确性、透明度、惊喜度更加感兴趣
支持该应用的设备	主要的固定设备(该应用只有一部分有限的功能在移动设备上是可以用的)
用户数	单用户应用
应用基础设施	基于浏览器的应用，具有两个内容分发模式：拉取模式(pull mode)(个性化搜索)和推送模式(push mode)(多媒体新闻摘要的主动分发)
屏幕实际尺寸	无限制

## 10.5.2.2 不同模型与设计约束条件之间的联系

在 MESH 工程中，推荐系统环境研究表明了每个模型(见表 10.5 和表 10.6)的重要特征，彼此之间是如何相互影响，以及哪种特征对过滤/推荐算法的选择、推荐系统的基础设计和适应推荐选择项(见图 10.3)有直接的影响。这种模式在我们分析系统关键特征的重要部分，并当其中一个模型发生变化时帮助确定对系统约束和要求的影响。

表 10.5 用户模型实例

人口统计学信息	只考虑工作作为具有歧视性的人口因素：我们将专业用户(如记者、摄影师等)和最终用户区分开
目标的存在性和本质	在拉取模式中，显性的目标通过用户在搜索引擎的用户查询表示；在推送模式中，最终用户除了通知他们喜欢的栏目和头条新闻没有其他目标，专业用户需要从拉取模式中，他们的阅读项目来检测和推断当前任务潜在的目标
期望水平	中等：即使推荐系统不能工作或者性能低，用户仍然能够使用该系统；推荐系统不是用户的活动中心
期望随着时间变化	是：当逐渐发现个性化搜索功能的好处时，用户的期望应该增加
用户环境的重要性	高：例如，用户在家、在工作或者假期不会阅读的同类的新闻
社交环境	独自的：新闻消费生来就是一个个体化的活动
信任和隐私问题	没有考虑(假设用户的利益和个人信息和数据的泄露之间做有必要的权衡)

表 10.6 数据模型实例

数据类型	半结构化的：新闻物品通过一系列的类别和概念来描述；应用了一些用来从物品(文本、语音、视频)的非结构化部分提取元数据的方法
元数据的质量	总的来说，中等：通过内容的语义标注具有很好的表现力，但是通过半自动提取的很多元数据不可靠
基于标准的描述	是：使用基于 IPTC 的本体论
项目的数量和多样性	数量巨大的新闻物品，覆盖了大量的各种主题
项目分布	长尾和主流：概念分布广泛，从像奥巴马选举这样的主流事件到那些发现新的青蛙物种相关的新闻
项目的稳定性和持久性	新闻数据集是不持久的，因为它是一个连续的数据流，所以要考虑推荐的新闻物品的不断变化(通常超过一定天数的新闻要忽略掉)
用户评分	显式评分(如星级评分)和隐式评分(通过监控在新闻物品的单击次数和花费的时间)都考虑了

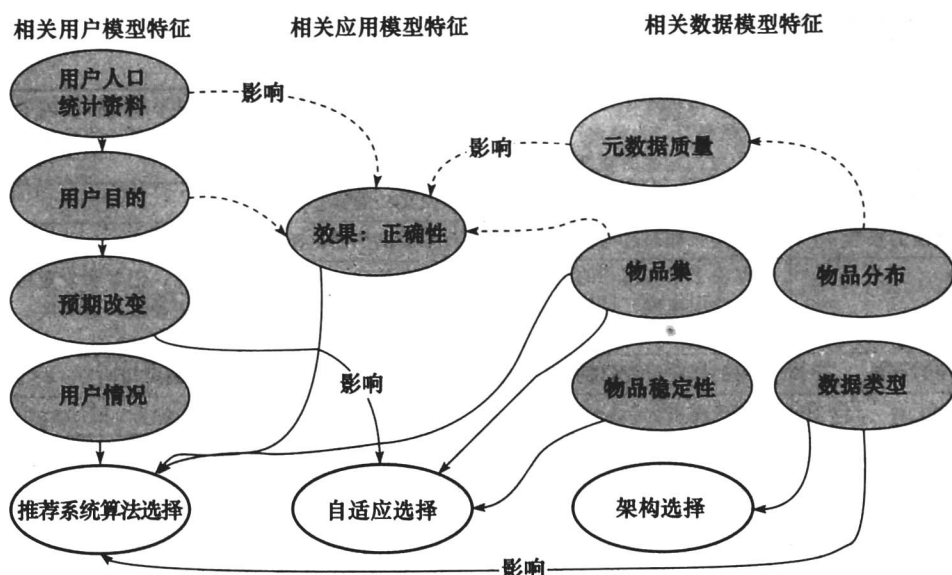


图 10.3 MESH 中的模型特征的依赖和影响图

**对推荐算法的影响：**在 MESH 中，推荐的最初目的是提高系统的性能；因此选择的算法必须克服基于内容的推荐和协同过滤推荐的局限性[7, 14, 1]。利用混合推荐算法[14]能够克服这些局限性。此外，不同类别的用户在一起，根据两种不同的模式(推/拉)，具有不同的目的和互动作用。在拉取模式中，推荐是以用户的查询为驱动而产生的，因此偏向于使用基于内容的方法，然而在推送模式中，对于最终用户，可能对协同过滤方法带来的更好的多样性感兴趣。因此，结合基于内容的方法和协同过滤相结合的混合方法要比其中单一方法更合适。另一个争论是关于元数据的：尽管有元数据(这可能导致使用基于内容的方法)，但是他们的质量仍然是中等的，即使不考虑推荐算法本身的质量，这都可能导致中等质量的推荐。因此不能只单靠基于内容的方法。

一旦决定结合不同类型的推荐，那么设计者必须对环境因素进行分析，选择更高准确率的方法。

- **协同过滤的方法：**研究数据表明不大可能使用经典的基于物品的协同过滤[45]，因为物品有非常大的量和动态性(新闻每隔几分钟都会发生)，对具体的新闻条目不可能有大量的用户评分(不管隐式的还是显式的)。这种特别严重的稀疏性要求设计者设想其他的协同过滤方法，例如，在基于用户的协同过滤中计算用户相似度时利用用户特征等[15]。
- **基于内容的方法：**高度结构化的元数据(用本体概念表达)，允许我们使用更加复杂的推荐方法，利用高度结构化的元数据(用本体概念表达)，可以提高推荐的准确率[17]。

最后，另一个重点是确定如何把这个算法结合在一起。基于环境研究，有多种启发式方法是可选的：基于用户类型(最终用户和专业人士)或者基于应用模式(拉取模式与个性化推送模式)的静态或者动态融合。用户的上下文也可以作为参数应用在不同算法及它们的融合方案中。

此外，在新闻背景下，对相似内容的考虑是必不可少的(例如，有关奥巴马选举的不同来源新闻内容)；因此为了避免推荐重复的内容，需要采用聚类附加机制。在这种情况下，把用户通常区分两个相似新闻物品的标准融合到整个推荐过程中是非常重要的。

要的。

**构建一个可适应的推荐系统的策略：**新闻数据研究表明新闻物品(新的新闻物品快速产生)和主题变动频繁。由于新闻的高波动性和高覆盖面的特性,对用户的研究证明用户不能完全表达他们对相关新闻的兴趣,所以用户信息也有一定的变动性。此外,在这个系统中,用户可以自由地浏览他们感兴趣的新闻。必须把用户感兴趣的新闻主题考虑在用户模型中,并保持模型的实时更新,以便可以将其应用在个性化新闻推送的过程中。用户模型随时间的自动和快速进化,对于一个可适应的推荐系统是非常必要的[38]。

**对系统架构的影响：**一开始, MESH 作为一种新应用, 对推荐系统有两种选择: 1) 分布式方法; 2) 完全集中式方法。分布式方法有以下优点: 很好的隐私管理和应用的独立性; 同时也存在许多严重的缺陷: 在不同的设备使用应用时, 需要在设备间同步用户配置文件及数据转换, 因此后端成本会非常高。因此, 决定选择集中式系统, 它能够简化推荐与系统的其他部分(如内容检索和内容整合)的集成。

### 10.5.3 实践: 模型的迭代实例化

由于推荐系统的构建是平行于系统的其他模块的, 所以不可能直到我们获得真实的数据才开始构建推荐系统。所以, 决定在公开数据集上构建算法进行评估:

- 对于**推荐算法**: 所有的测试在 MovieLens 和 IMDB 数据集上完成;
- 对于**用户信息自适应算法**, BARB<sup>⊖</sup>数据。选择此数据集的原因在于需要在内容消耗上进行时间进化。

结果表明在 MESH 中开发的混合推荐方法要比目前先进方法(基于内容和协同过滤)提供更加准确的结果。在这种情况下, 元数据的质量接近我们的期望要求, 并且用户上下文也没有被考虑进来。因此, 在 BARB 数据集(6 个月时间内有关电视节目消费量的数据集)上进行实验时, 我们不由自主地创造了一个新的数据模型和一个稍有变化的用户模型:

- **数据模型**: 物品是用有限的元数据描述的电视节目, 其中元数据中包括节目的高层次类别以及从节目描述文档中提取的关键词。物品的分布是带有从低级到中级的注释质量的主流分布。物品的稳定性也不相同, 由于电视节目可以适用于很长的一段时间, 所以我们可以把它们看成相当稳定的(电视具有明显的重复性, 但是新闻节目不能每天都重复)。该数据集改变了影响推荐性能的两个特征: **元数据的质量和物品的分布**, 影响适应性结果的一个特征也被改变了: **物品的稳定性**。
- **用户模型**: 在 BARB 数据中没有专业用户, 因此没有与特殊目的相关联的推荐。对推荐的准确率具有主要的影响。

BARB 本可以帮助我们评估用户偏好学习算法, 但是经过一周的学习结果表明用户特征仍然是稳定的, 也就是说偏好学习机制对用户偏好并没有多大影响<sup>[38]</sup>。因此不仅需要根据算法自身的运行状况解释我们的实验结果, 还要从算法的行为和所选择的数据集的特征来解释实验结果。整个算法本来是用来处理大量的物品的, 非持久的、并且带有高质量元数据的主流长尾分布。假设算法是非常被动反应的, 而且建立在基于物品的平均消费极大优于 TV 节目的每天消费之上。通过这个实验, 我们认识到改变推荐环境(数据、用户和应用模型)对预期结果、过滤算法的推荐解释及其他与推荐系统相关的机制都有巨大的影响。

⊖ BARB-Broadcasters' Audience Research Board, <http://www.barb.co.uk>



因此,我们尽可能利用与 MESH 系统中相似的数据。利用一个称为“New@Hand”的工具启动新的实验。New@Hand 结合文本特征和协同信息为最终用户提供新闻推荐。通过 RSS 反馈从多种在线新闻服务中对新闻项目进行自动定期更新。新闻的主题、摘要和类别用系统本体概念进行标记。在三周时间内,利用主观定性评分方法(见 10.4.2.3 节),分析了偏好学习算法对 20 个用户性能(为了比较算法的不同配置的学习性能,将用户分到不同小组)。由于在此实验中的数据模型与 MESH 中的相似,用户模型有轻微的不同,得到了基本相似的实验结果。

通过数据、用户和应用模型的实例化,对推荐环境进行详细的研究,并区分模型特征的相关性和影响力,为了避免错误和在死角上浪费时间,我们认为在设计阶段必须解决掉这些关键问题。

## 10.6 总结

本章中说明了尽管有多种不同的可用技术能够用于实现一个推荐系统,但是往往很难进行比较,一个系统的环境评估是做出适当选择的关键。为此我们构建了对推荐系统设计有重要影响的环境特征的三种模型:应用、用户和数据。构建这三种模型和研究它们之间的相互作用,可以对适当的推荐算法、系统结构和用户模型进行优化选择。

在实际中(如我们演示的实例),这三种模型不能一次性地完成一个完美的设计。但是它们对迭代设计方法是非常有用的支持,这是超越卓越的技术以及实现最终的目标的最佳方式:用户满意度。

## 参考文献

1. Adomavicius, G., Tuzhilin, E.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
2. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, vol. 22, pp. 207–216. ACM (1993)
3. Ali, K., van Stam, W.: Tivo: making show recommendations using a distributed collaborative filtering architecture. In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. Seattle, WA, USA (2004)
4. Anand, S.S., Mobasher, B.: Contextual recommendation. In: *From Web to Social Web: Discovering and Deploying User and Content Profiles*, Lecture Notes in Computer Science, pp. 142–160. Springer-Verlag (2007)
5. Anderson, C.: *The long tail*. Wired (2004)
6. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. In: *Applied Artificial Intelligence*, pp. 687–714. Taylor and Francis (2003)
7. Balabanović, M., Shoham, Y.: Fab: content-based, collaborative recommendation. *Communications of the ACM* 40(3), 66–72 (1997)
8. Bernhaupt, R., Wilfinger, D., Weiss, A., Tscheligi, M.: An ethnographic study on recommendations in the living room: Implications for the design of itv recommender systems. In: *EU-ROITV'08: Proceedings of the 6th European conference on Changing Television Environments*, pp. 92–101. Springer-Verlag (2008)
9. Bias, R., Mayhew, D.: *Cost-Justifying usability*. Morgan Kaufman Publishing (1994)
10. Bolger, N., Davis, A., Rafaeli, E.: Diary methods: Capturing life as it is lived. In *Annual Review of Psychology* 54(1), 579–616 (2003)
11. Bonnefoy, D., Bouzid, M., Lhuillier, N., Mercer, K.: More like this or not for me: Delivering personalised recommendations in multi-user environments. In: *UM'07: Proceedings of the 11th international conference on User Modeling*, pp. 87–96. Springer-Verlag (2007)
12. Bonnefoy, D., Drgehorn, O., Kernchen, R.: *Enabling Technologies for Mobile Services: The*

- Mobilife Book, chap. Multimodality and Personalisation. John Wiley & Sons Ltd (2007)
13. Bonnefoy, D., Picault, J., Bouzid, M.: Distributed user profile. Patent applications EP1934901, GB2430281 & WO2007037870 (2005)
  14. Burke, R.: Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* **12**(4), 331–370 (2002)
  15. Cantador, I.: Exploiting the conceptual space in hybrid recommender systems: a semantic-based approach. Ph.D. thesis, Universidad Autónoma de Madrid (UAM), Spain (2008)
  16. Cantador, I., Bellogn, A., Castells, P.: News@hand: A semantic web approach to recommending news. In: *Proceedings of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2008)*, *Lecture Notes in Computer Science*, vol. 5149, pp. 279–283. Springer-Verlag (2008)
  17. Cantador, I., Fernández, M., Vallet, D., Castells, P., Picault, J., Ribière, M.: Advances in Semantic Media Adaptation and Personalization, *Studies in Computational Intelligence*, vol. 93, chap. A Multi-Purpose Ontology-Based Approach for Personalised Content Filtering and Retrieval, pp. 25–51. Springer (2008)
  18. Cantador, I., Szomszor, M., Alani, H., Fernández, M., Castells, P.: Enriching ontological user profiles with tagging history for multi-domain recommendations. In: *Proceedings of the 1st International Workshop on Collective Semantics: Collective Intelligence and the Semantic Web (CISWeb 2008)*, pp. 5–19 (2008)
  19. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: Gate: A framework and graphical development environment for robust nlp tools and applications. In: *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)* (2002)
  20. Dong, J., Martin, S., Waldo, P.: A conference on human factors in computing systems input and analysis tool for information architecture. In: *Conference on Human Factors in Computing Systems*, pp. 23–24 (2001)
  21. Duda, R.O., Hart, P., Stork, D.G.: *Pattern Classification*. John Wiley, New York (2001)
  22. Gilb, T.: *Principles of software engineering management*. Arron Marcus Associates (1988)
  23. Ha, S.H.: An intelligent system for personalized advertising on the internet. LNCS 3182 pp. 21–30 (2004)
  24. Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. In: *CSCW'00: Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pp. 241–250. ACM, New York, NY, USA (2000)
  25. Herlocker, J.L., Terveen, L.G., Konstan, J.A., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Transactions on information systems* **22**, 5–53 (2004)
  26. International Organisation for Standardisation (ISO): ISO 13407: Human centred design processes for interactive systems.
  27. Jameson, A., Baldes, S., Kleinbauer, T.: Two methods for enhancing mutual awareness in a group recommender system. In: *AVI'04: Proceedings of the working conference on Advanced visual interfaces*, pp. 447–449. ACM, New York, NY, USA (2004)
  28. Lhuillier, N., Bonnefoy, D., Bouzid, M., Millerat, J., Picault, J., Ribière, M.: A recommendation system and method of operation therefor. Patent application WO2008073595 (2006)
  29. Lhuillier, N., Bouzid, M., Gadanho, S.: Context-sensitive user preference prediction. Patent application GB2442024 (2006)
  30. Lhuillier, N., Bouzid, M., Mercer, K., Picault, J.: System for content item recommendation. Patent application GB2438645 (2006)
  31. Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction* **14**(1), 37–85 (2004)
  32. McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P.: Cats: A synchronous approach to collaborative group recommendation. In: G. Sutcliffe, R. Goebel (eds.) *Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference*, pp. 86–91. AAAI Press (2006)
  33. McNee, S.M., Riedl, J., Konstan, J.A.: Accurate is not always good: How accuracy metrics have hurt recommender systems. In: *CHI'06 extended abstracts on Human factors in computing systems*, pp. 1097–1101. ACM, New York, NY, USA (2006)
  34. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.* **7**(4) (2007)
  35. Mobasher, B., Jin, X., Zhou, Y.: Semantically enhanced collaborative filtering on the web. *Web Mining: FromWeb to SemanticWeb* pp. 57–76 (2004)
  36. Oppenheim, A.: *Questionnaire Design, Interviewing and Attitude Measurement*. Continuum International Publishing, New York (2001)

37. Papadogiorgaki, M., Papastathis, V., Nidelkou, E., Kompatsiaris, Y., Waddington, S., Bratu, B., Ribière, M.: Distributed user modeling for personalized news delivery in mobile devices. In: 2nd International Workshop on Semantic Media Adaptation and Personalization (2007)
38. Picault, J., Ribière, M.: An empirical user profile adaptation mechanism that reacts to shifts of interests. Submitted to the 18th European Conference on Artificial Intelligence (2008). <http://www.mesh-ip.eu/upload/ecai2008.pdf>
39. Picault, J., Ribière, M.: Method of adapting a user profile including user preferences and communication device. European Patent EP08290033 (2008)
40. Ribière, M., Picault, J.: Progressive display of user interests. Tech. rep., Motorola (2008). <https://priorart.ip.com/download/IPCOM000167391D/>
41. Ribière, M., Picault, J.: Method and apparatus for modifying a user preference profile. Patent application WO2009064585 (2009)
42. Rich, E.M.: User modeling via stereotypes. *Cognitive Science* 3, 329–354 (1979)
43. Rokach, L. and Maimon, O. and Arbel, R., Selective voting-getting more for less in sensor fusion, *International Journal of Pattern Recognition and Artificial Intelligence* 20(3):329–350 (2006)
44. Rugg, G., McGeorge, P.: The sorting techniques: a tutorial paper on card sorts, picture sorts and item sorts. *Expert Systems, The journal of Knowledge Engineering* 14(2), 80–93 (2002)
45. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW'01: Proceedings of the 10th international conference on World Wide Web, pp. 285–295. ACM, New York, NY, USA (2001)
46. Shepherd, M.: Tutorial on personalization and filtering on the web. Web Information Filtering Lab, Dalhousie University, Canada. <http://ncsi-net.ncsi.iisc.ernet.in/gsdll/collect/icco/index/assoc/HASH011b.dir/Tutorial-Shepherd.ppt>
47. Shoval, P., Maidel, V., Shapira, B.: An ontology-content-based filtering method. *International Journal of Information Theories and Applications* (15), 303–318 (2008)
48. Sieg, A., Mobasher, B., Burke, R.: Ontological user profiles for personalized web search. In: Proceedings of AAAI 2007 Workshop on Intelligent Techniques for Web Personalization, pp. 84–91. Vancouver, BC, Canada (2007)
49. Signa, R.: Design strategies for recommender systems. UIE Web App Summit. <http://www.slideshare.net/rashmi/design-of-recommender-systems>
50. Snyder, C.: Paper prototyping: The fast and easy way to design and refine user interfaces. Morgan Kaufmann Publishing, San Francisco (2003)
51. Terveen, L., Hill, W.: Human-Computer Interaction in the New Millennium, chap. Beyond Recommender Systems: Helping People Help Each Other, pp. 487–509
52. Villegas, P., Sarris, N., Picault, J., Kompatsiaris, I.: Creating a mesh of multimedia news feeds. In: European Workshop on the Integration of Knowledge, Semantics and Digital Media Technologies (EWIMT), pp. 453–454. IEE (2005)
53. Yu, Z., Zhou, X., Hao, Y., Gu, J.: Tv program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction* 16(1), 63–82 (2006)
54. Zhang, M., Hurley, N.: Avoiding monotony: improving the diversity of recommendation lists. In: RecSys'08: Proceedings of the 2008 ACM conference on Recommender systems, pp. 123–130. ACM, New York, NY, USA (2008)
55. Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: WWW'05: Proceedings of the 14th international conference on World Wide Web, pp. 22–32. ACM, New York (2005)

# 匹配推荐系统的技术与领域

Robin Burke 和 Maryam Ramezani

**摘要** 推荐系统涉及的技术和方法极其广泛。本章旨在帮助研究人员和开发者分辨各种推荐技术最适合应用的推荐领域。不同于其他对推荐系统的分类，本章的关注点在于知识问题。例如，推荐系统工作需要知识的种类，以及这些知识的来源。不同的推荐领域（以图书和公寓为例）将提供不同的收集和应用知识的机会。基于上述内容产生了如何将领域特点和推荐技术结合的问题。

## 11.1 简介

不同于其他软件系统，如统计计算机软件包或地理信息系统那样存储和使用一类特定的数据，推荐系统被定义为一种特殊类型的程序。它被定义为一种与用户交互的系统，具体指：“推荐系统是任何以产生个人推荐作为输出的系统，或者是用个性化手段对用户的偏好产生引导的系统，或者是在大量可能选择性空间中提供有效参考的系统。”[11]这一宽泛的定义使得推荐系统的研究领域极为广泛，但是对开发者而言却很难提供指导。由此产生了一个关键的问题，即如何有针对性地应用某些技术去解决相应的推荐问题。本章将重点阐述如何解决该问题。

## 11.2 相关工作

针对推荐系统的分类方式有很多。Burks[11]的论文区分了 5 类主要的推荐技术，包括：协同过滤、基于内容的推荐、基于效用的推荐、基于人口统计的推荐和基于领域知识的推荐。论文讨论了每种技术的优缺点，并且为减少使用单一技术带来的弊端，论文中提出了混合推荐系统以提高系统性能。他认为，相比于单一的推荐技术，在有效降低召回率的同时，混合推荐系统可以取得更好的推荐效果。推荐系统的早期工作[54]主要聚焦于协同过滤推荐算法，协同过滤将推荐系统划分为 5 个确定的维度，并且在这些维度空间设计推荐系统。维度描述了用户与推荐系统交互的属性以及用户评价的聚集程度。Konstan 和 Schafer[61]提出了一种基于电子商务应用的协同过滤推荐技术的分类方法。其将相应推荐分为三个大类：功能性输入输出、推荐方法以及其他设计需要。例如，个性化程度和传递方式。下面提出一种基于双主要准则的八维度推荐系统分类方式：用户配置文件的生成与维护，以及用户配置文件的开发技术。按这种分类方式，传统的推荐系统模式被抽象为在相同领域下对系统的分析。最新的关于推荐系统[3]的综述将推荐方法分为基于内容的推

---

Robin Burke, Center for Web Intelligence, College of Computing and Digital Media, 243 S. Wabash Ave., DePaul University, Chicago, Illinois, USA e-mail: rburke@cs.depaul.edu

Maryam Ramezani, Center for Web Intelligence, College of Computing and Digital Media, 243 S. Wabash Ave., DePaul University, Chicago, Illinois, USA e-mail: mramezani@depaul.edu

翻译：周俊宇，吉林大学-丁彬钊 审核：郭晓，张彤彤，吴亮

荐、协同过滤推荐以及混合推荐三类，并且在每类中将推荐器具体划分为基于启发式的算法和基于模型的算法两种。

该工作与先前的分类方法不同，其不再以特定的兴趣维度作为划分现有推荐系统的标准，而是一种以人工智能为核心的探索方法，关注于推荐系统所需知识的来源以及加诸其上的各类限制条件。本章讨论了如何将不同的推荐技术应用于不同类型的实际问题，以期在推荐技术选择时给予指导意见。从内容本身角度而言，本章起到为推荐系统开发者提供技术指导的作用。

11.3 知识源

对一个人工智能系统开发者而言，在开始工作之前首先确定系统用到的知识来源和知识类型。就推荐系统而言，应该有两项主要基本知识：推荐系统是个性化的，因此必须包含关于用户的知识；还必须包含关于所推荐物品的特征信息。

不需要物品特征信息和领域知识被认为是协同过滤的主要优势之一，而其他的推荐技术需要特征信息。

对于具体的推荐实例，我们面对的是一个特定的用户并且争取为他提供个性化推荐。这种情况下，我们将关于用户的知识分为两类：关于目标用户的知识和关于整个用户群的知识。因此推荐过程中将会用到三大类知识：

- 社群：关于整个用户群体而不是目标用户的知识。
- 个人：关于目标用户的知识。
- 内容：关于将要被推荐的物品的知识，以及泛指其用途方面的知识。

Felfrening 和 Burke 在文献[18]中介绍了一种推荐所需知识的分类方法。图 11.1 展示了这种分类：进一步细化每个类别的子类型的知识，所有的这些知识都已经被应用在一些现有的推荐系统中。下面将详细介绍这些子类型知识。

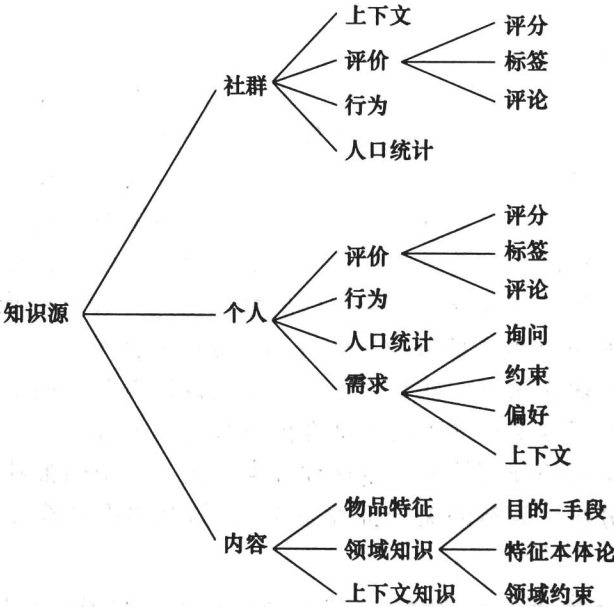


图 11.1 推荐系统中的知识源分类

社群知识是系统中存储的所有用户信息的总和。通常通过最简单的用户文件类型，协同过滤推荐提升了社群知识的使用，比如，用户评价分为喜欢或不喜欢。该方式常用在如



MovieLens 和其他知名的推荐网站中。当然其他类型的知识也会产生一定的作用,在 I-SPY 协同过滤搜索应用网站中,网站会记录那些与用户选择项(选择链接)相关联的用户查询词(请求)[63]。基于用户的人口统计信息也被应用于推荐系统中。

个人知识通过推荐系统主导了特定的用户交互。个人知识在推荐的过程中有两种情况,一种情况是把用户的过去特定浏览记录作为推荐的启动数据(这种情况下语义相对是模糊的),另一种情况是在推荐开始之前或过程中用户自己提供他们的兴趣偏好(这种情况下语义是相对明晰的)。除了行为、评价和针对社会范畴的人口统计学知识,个人的需求更容易被发现。举例来说,评论系统是一种基于知识的推荐系统。一个用户对于推荐物品的测试以及对于多维评论的回馈将对下一轮检索的偏好产生约束(且/或)[12]。这种系统与用户的交互常常始于用户所表达的询问。

内容知识有多种不同形式。在最简单的情况下,系统仅有关于将要被推荐的物品特征的知识。这些知识可以使系统学习用户偏好的特征。领域知识往往表示更加复杂的内容知识概念。如手段-目的知识,这个知识是针对这种情况,即用户所考虑什么样的手段/特征适用于什么样的目标/结果。由于特征本体让各个特征相互产生关联,所以相似度和相异度才可以被合理地评估。域完整性约束在防止推荐系统推荐出与领域要求不一致的物品时起到非常重要的作用。

在很多应用领域中,上下文是一个很重要的因素。上下文往往在产生一个合适的推荐中具有重要的作用。举例来说,如果你想要找一家饭店举行商务会谈和要找一家饭店进行自己孩子四岁的生日聚会,这两件事的推荐将会有很大的不同(甚至用户请求和用户文件信息可能相同)。在不同的推荐算法研究(包括协同过滤算法)中基于上下文的研究一直是热点[2, 53, 31, 5, 62, 70]。然而,如何最大限度地发掘上下文给推荐带来的好处甚至是如何定义相关的术语都是学界探讨的问题。本节将只介绍部分关于上下文在推荐系统中的内容。

### 11.3.1 推荐类别

在本书的其他章节,推荐类别已经阐述得很详细了。但是,结合本节主题知识源,关于推荐类别的内容还是值得继续讨论的(见图 11.2)。

- 协同过滤推荐是用个人知识与同类型的社群知识进行匹配,然后从他或她的同类上推断出他的偏好。通常,在协同过滤中,个体需求是不使用的,或者被简单地用作过滤器。
- 基于内容的推荐从另一个方面来说更加关注于个人的内容,使用物品的特征和用户的评价去训练分类器,可以使分类器在对用户推荐新物品的时候依据他们的偏好。
- 基于知识的推荐是一个多面手,它几乎使用了所有的领域知识而不仅是物品特征。

图 11.2 显示了知识源和推荐类别的关系。

如果用透视的方法来看知识源,那么混合推荐结合了多种知识源,与上面所讨论过的传统推荐类别有所不同。当我们需要把只能适用于一种知识源的推荐算法改造成可以适用于其他相关的知识源的推荐算法的时候就需要用到混合推荐。

当然,知识源本身不能构成推荐系统,因为一个人工智能系统需要算法。但是推荐系统难以归纳完全,因为新的推荐算法总是以极强的形式化方法给出的。不过,从大体上来说,推荐系统可以归纳为如下几类:协同过滤推荐系统使用多种分类算法为了解决高维数据空间高度稀疏的情况;基于内容的推荐系统使用二位学习型算法,它适用于低维数据空间;基于知识的推荐系统使用多种体系的推理机制。

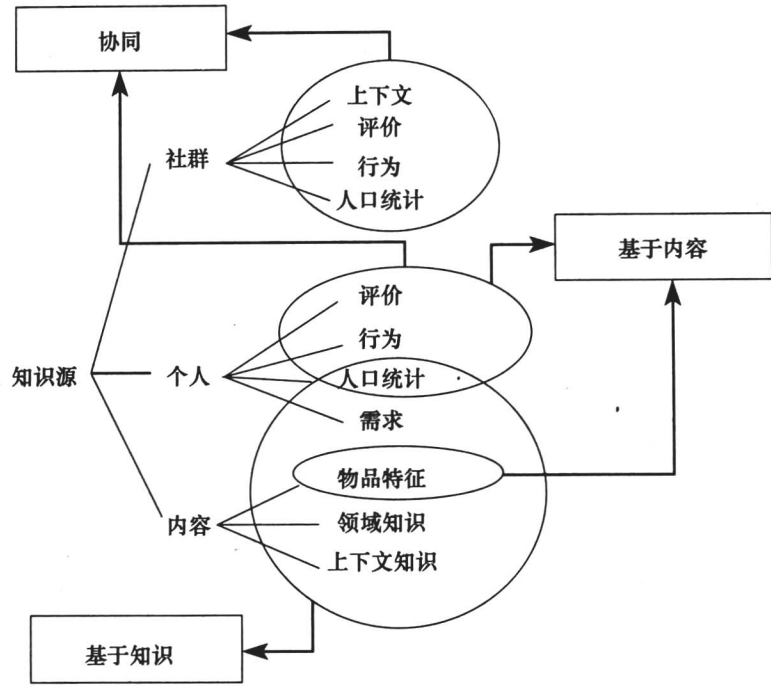


图 11.2 知识源和推荐类型

但是，一种算法只有找对了知识源才能发挥作用。如何找对知识源是本章的主题。通过总结不同的应用领域以及不同的用户交互方式，我们可以得出不同知识源的适用范围以及特征。这些结论反过来也可以作为选择合适的推荐算法的指导。接下来我们将研究领域的特点。

11.4 领域

推荐领域是推荐系统将要操作的一组物品，也包括推荐系统想要支持的一些目的和手段。下面举一个针对专门的推荐系统的例子，帮助政府情报分析员鉴别新闻报道的新闻推荐系统与通用新闻推荐系统，如 Google News，在实现上的考虑会有所不同。反过来，领域的特点影响了不同知识源的可用性与实用性。以在线新闻系统为例，不是一小部分而是大部分的新闻用户没有时间去查看和评论。更进一步，新闻本身也经常发生变化。所以我们可以归纳出这样一条结论——“社群/评价/评分”知识源是非常稀疏和动态的。

领域的另一个方面与推荐系统所植入的广泛应用有关。如果推荐系统只是一个大系统的一部分，如电子商务网站，那么推荐系统在日常用户交互中所产生的影响越小越好。这也许会限制用户需求收集的程度。从另一个方面来说，如果推荐系统是用用户所使用的主要应用，那么推荐系统可以更直接地获得用户数据。

我们发现了实现者需要考虑的有关领域的 6 大重要特征：异构性、风险性、变动性、交互风格、偏好稳定性和可理解性。

11.4.1 异构性

一个相同性质(简称“同质”)的物品组成的空间包含很多拥有不同特征的物品，而且，最重要的是它们可以满足不同的目标。举例来说，一个电子商务网站如亚马逊，拥有大量的同质物品需要推荐。甚至在图书这样一个简单的推荐类别中也包含诸多不同的类别，如

家居维修、爱情小说、厨艺、童话故事等，都存储在同一个数据库中。

一个单一性的推荐空间意味着，与领域相关的内容知识将可以更方便地获取与维护。考虑这样的情况，一个仅推荐数码相机的网站，而与之相对的是推荐所有电子产品的网站。数码相机网站将会有足够的精力去使用摄影方面的专业知识，但是通用电子产品网站将会面临一个极具挑战性的任务，为了给所有电子产品消费者推荐而构建一个知识工程。如果物品之间的特点变化很大，那么即使一个简单的关于物品的特征的目录也很难设计。

#### 11.4.2 风险性

推荐领域可以通过在用户产生推荐时的风险程度不同来区分。一个价值 99 美分的音乐专辑的风险是低的，但是一个价值 150 万美元的财产所有权或者医疗诊断的风险就很高了。风险性决定了用户对于错误推荐的忍耐程度。在某些领域中，错误的负面影响可能会非常显著。比如，在一些选择中可能会存在高额的消费和高风险的问题。

从另一个方面来考虑风险领域，推荐系统所依赖的有效解决方案存在很多约束条件。举例来说，一个房地产购买者，对于地理位置、价格、舒适度有很高的约束条件。综上所述，物品的风险度越高，用户对于错误的容忍度就越低。

#### 11.4.3 变动性

推荐系统应用在这样的一些领域中，要推荐的物品拥有很长的使用周期，如图书。但是有时推荐系统也会应用在其他的一些领域当中，待推荐物品的价值和相关度只有一个较小的使用周期，如新闻报道。一个高度混杂的领域是由于内部构成的物品经常发生变化。

在这样的领域中，推荐系统面临着物品需要源源不断地被整合到自身的数据源当中的问题。因此极大地增加了任何种类的选项数据的稀疏性，因为新的物品必然只被很少的用户查看。已经存在了一段时间的物品可能积累了一些评分信息，但是随着时间的流逝它们将不再具有相关性。

#### 11.4.4 交互风格

在系统中，用户不遗余力地与推荐系统进行交互，并且此时系统从用户不明确说明的偏好中挖掘出用户的行为。例如，当用户浏览网站的时候，他会在服务器上留下一些浏览的行为记录，而这些记录将会作为推荐的依据。隐含的输入可能包括用户正在浏览的特殊物品、用户的交易历史、流逝的时间、购物车/消费习惯等。明确的输入包括用户提出一个意见或者请求或者将个人信息添加到系统中。这里必须有一些用户能感觉到的好处才可以盖过这样做的用户代价。

隐含的输入天然就存在噪声，因为它来自对用户行为的推断。虽然一些研究人员已经可以从隐含输入中挖掘出偏好甚至领域知识[58, 72]，但是隐含输入还是只适合收集简单的评分知识。明确的输入可能会造成稀疏性增加，因为产生这些输入的代价会使得用户很少这样做。

#### 11.4.5 偏好稳定性

用户的偏好依据持续时间的不同会有变化。举例来说，一个购买了数码相机的用户可能在购买之后变换他的偏好，因为他不会对购买完成的物品再产生偏好。但是一个对喜剧电影感兴趣的用戶可能希望长时间地获得喜剧电影的推荐。当然，对于一些物品的偏好会增加也会随着

时间衰减。举例来说,当一个用户喜欢的篮球队正在参加一项重大的赛事,用户对这支球队的兴趣将会很高。然而当这支球队被踢出局或者比赛结束,用户的兴趣将会发生转移。

稳定的偏好意味着过去收集的用户意见数据同样适用于今天。因此使用该数据将会使获得高质量的知识源变得容易。不稳定的偏好数据意味着过去被收集的数据可能大打折扣或不再适用。这就使得搜集当前用户特定需求变得重要。如果用户在一个会话周期内产生大量隐语义数据,那么使用个人会话作为数据文件就变成一种可能。

用户偏好不稳定的问题可以通过收集更多的数据来改进。如果一个用户可以用单个会话周期内产生足够的数据去代表他的偏好情况,那就没有必要通过历史数据挖掘偏好,并且也没有必要提高用户偏好的稳定性。这种情况在网络的一些个性化应用中很常见。当用户浏览网站会产生大量的点击事件时,这将产生足够的隐语义数据供推荐系统使用。

#### 11.4.6 可理解性

某些应用(举例来说,通常具有高风险)可能需要系统解释它的推荐。比如,要回答这样一个问题“为什么要推荐这个物品给我?”的解释增强了用户对推荐系统的信任,使他们认为这样一个推荐是合适的[42],并且也使得推荐系统更容易被用户接受。

当使用基于内容的知识源时,解释推荐是很直接的。解释基于社群知识的推荐将有很多挑战[41, 57]。请参看 Herlocker 等的论文[26],该论文阐述了针对一些可选择的事物的评估。

### 11.5 知识源

如同我们上面所阐述的那样,领域的选择和应用场所的特点明确地限制了推荐系统所依赖的知识源的种类。相应地,知识源的质量和可用性影响了一个推荐系统可以有效利用的推荐技术。

#### 11.5.1 社群知识

社群知识使得协同过滤算法使用用户的邻居的偏好推断用户的个人信息来进行预测成为可能。如果要给这种知识建模,评分是一种最直接的数据类型。评分的知识常被描述为一个  $m$  行  $n$  列的矩阵。 $m$  代表的是用户数, $n$  代表的是物品数。每一个与之相符的输入项就是用户对某一个物品的评分。基于模型的技术使用这个矩阵提前产生模型,然而基于内存的技术在推荐的时候使用矩阵产生推荐。

使用其他类型的用户意见数据是一个研究热点。用户标签是意见数据的主要来源[49, 23, 69, 40, 68]。在这种情况下,数据将主要分成三个部分:用户、物品、标签。评论形式的文本数据也已经被研究[1]。针对当前的协同过滤推荐算法,上述提到的这些多维数据源就对现有的协同过滤算法提出了挑战[2]。

在异构的领域当中,社群知识应作为一种数据源,社群知识的采集通过用户的输入,并且构建社群知识不需要额外的知识工程。然而,对于高风险领域或需要进行解释的领域,社群知识不够精确和可靠。

社群知识将会变得越来越稀疏并且异构性越来越强。对于高变动性领域,社群知识会变得越来越稀疏。在物品产生消逝变化非常快的情况下,指定的任何用户可以评论指定的任何物品的概率就会减小。由于其他稀疏性的影响,数据高变动性的问题将通过大量用户数量来改良。举个例子,谷歌新闻可以通过网站上的大量基础活跃用户使用协同过滤进行推荐,甚至在高度变动的新闻领域也是可行的[16]。

对隐式交互类型的领域来说,采用社群知识是合适的,因为这样有可能采用协同过滤中机器学习和统计技术的经典算法挖掘用户行为。在用户偏好不稳定的领域,社群知识可能会产生误导,因为历史数据都是不可信的。

### 11.5.2 个人知识

对于推荐系统,个人知识对个人推荐是十分必要的。

在协同过滤推荐中,与目标用户的个性化知识相匹配的是来自整个用户人群的社群知识。针对个人最直白的一种说法是,个人推荐的知识源都是相同的类型,并且都需要某种相似度评判标准以备在个性化推荐过程中比较个体差异。但是研究人员在针对个人推荐的评分工作中也使用了人口统计知识。Krulwich[32]使用通过销售调研的人口统计组数据去产生关于一系列产品和服务的推荐。在其他推荐系统中,机器学习常被用作对于基于人口统计数据的分类器的训练[52]。

在异构领域中,把用户的输入转换成确定的推荐物品是困难的。举例来说,两个对于电影有相同偏好的用户对于音乐是否有相同偏好是不确定的。

在缺少社群知识的情况下,尤其是与评分有关的个人知识可以和与物品特征有关的内容知识相结合,以构造一个使用有监督分类学习的典型的基于内容的推荐系统[34, 47, 51]。

如果是一个需要用户短期需求的知识的领域,那么某类基于知识的推荐系统将非常适合它。查询是最基本的请求输入项,通常包括如下几种类型:用户的陈述,任何形式的系统的接收的内容,以及用户正在搜索的内容。由于约束条件和偏好的制约,用户只能在有限的情况下排列他们的选择。举例来说,一条狗的主人在租住公寓的时候必须考虑公寓能够接受他的宠物。有孩子的父母在选择公寓的时候通常喜欢靠近公园和操场。在有高风险和需要解释的推荐领域中,通常需要从用户那里提取需求和约束条件。同样,在一些用户偏好不确定的领域中用户的需求也是必需的,因为历史数据是靠不住的。

在很多推荐系统中,不止一种类型的用户输入项被使用。比如,在 CBR 推荐系统中使用用户的偏好和约束[43]。文献[15]中提到一个关于偏好发现的启发式方法。

### 11.5.3 基于内容的知识

最基本的内容知识是物品特征,物品特征的数据(例如,一些数值如价格或者象征性的标志(举例来说,像目的地或飞机场))可以通过产品数据库来查找。这些特征可以被推荐系统所使用。当然实施者希望可以约束特征空间。举例来说,一部电影的全体演员和职员表可以作为特征数据,但是也会伴随很多稀疏的特征,它们的利用率极为低下。如果整理这个列表挑选出最佳的演员、最佳的导演、最佳的编剧将会更加适用。

如果物品由非结构性的文件所描述,如新闻报道,那么开发者将使用信息提取技术(IE)去提取用户在推荐过程中的选择特征。标准化技术包括敏感词排除、抑制特征空间简化等。通过特征选择复杂特征的技术,如信息增益、交互信息、交叉熵、或者优势率[44],特征可以被进一步消减。高度结构化的文件,如 HTML 页面,可以针对特征提取提供更多的选择。在文献[30]中论述了如何应用 IE 技术提取结构化文件或半结构化文件的内容知识。在多媒体领域中内容的知识是一个新的挑战。Hauptmann 在文献[22]中讨论了多媒体检索技术。Osmar 等[71]研究了多媒体的数据挖掘,在论文中阐述了针对推荐的大量有用技术。

基于内容的推荐算法和基于知识的推荐算法所做出的推荐质量完全取决于它们所依赖的用户的选择的內容数据的质量。事实上,缺少可以信赖的物品特征常常被认为是使基于内容的推



荐失效的主要原因。创造并维护一个包含有用的物品的特征信息的数据库的开销是不应该被低估的,尤其是在混合推荐领域中。举例来说,在数码相机领域或手机行业,新的技术革命常常发生。这就要求针对每一个物品的所有个人推荐和与之相关的计划都要更新。如果存在大量通过多种渠道提取的没有完全独立的特征数据,那么系统将承受来自这些特征数据的噪声的干扰。从另一个角度来说,有高风险的一些应用将会更加关注如何获取干净的特征数据。总而言之,解决这一问题需要基于人工复查特征数据和基于人工的标签。

### 11.5.3.1 领域知识

基于知识的推荐系统所需要知道的不仅是物品所对应的特征。推荐系统所能使用的最基本的领域知识形式是物品的特征实体。这样的实体使得推荐系统能够推测出更深层次的特征和基本特征的相同或区别。举例来说,餐厅推荐系统 Entree[12]有包含各种烹饪类别的本体知识,并可以做出这样的推论:与德国餐馆相比,泰式餐馆更加类似于越南餐馆。

很多高风险选择给该领域的推荐系统强加了必须遵守的限制。举例来说,一个理财产品的推荐系统[19]也许会知道特定的投资设备仅适合于拥有特定特征的用户(比如,对年龄超过 55 岁的人来说一项特殊的生命保险政策并不适合他们)。所谓推荐的问题在某种情况下可以被完全地表述为如下形式,即将用户和系统正在提出的约束进行满足的过程。

最后的一种领域知识是方法-结果知识,这种知识可以使用户的目标(结果)与满足用户的产品(方法)对应起来。举例来说,一个要买相机的人可能不太了解相机,但是他知道他要在她女儿进行篮球赛时给她拍照片。用户从推荐系统获益的一个方面是:他们可以做出更好的选择而不用熟悉复杂的产品空间。

表 11.1 总结了这些领域及其对知识源的影响。

表 11.1 推荐领域对于知识源的影响

特性	社群	个人	内容
异质		可能会不充分地传递给未看到的物品	难以设计/维护
高风险	不够准确/可靠	通常会包含必要条件和限制	有领域限制
高变动	稀疏数据		
隐式		缺少具体要求	
偏好不稳定	历史数据不可靠	历史数据不可靠需要用户必要条件	
需要解释	解释不充分	必要条件能对应到物品	可以用到领域知识

## 11.6 从领域到技术

在讨论推荐系统领域之前需要先说一些基本的问题。首先,在一些领域中社群知识不是那么好用,比如,高风险领域和高变动的领域。在高变动领域中,一个物品可能没有足够时间在被其他物品替换掉之前建立起在大量用户中的声望。

当存在与领域相关的巨大风险时,除了其他用户喜欢这个物品这种简单理由,大部分用户需要对合适的推荐有更加合理的解释。第 25 章将论述关于协同过滤推荐系统的健壮性。如果我们优先考虑当前的上下文环境,那么通过用户过去交互记录所获得的用户信息也就不那么令人满意了。

同样,如果考察交互,那么我们会发现从每种类型的交互中提取知识并不总是那么容易。在推荐系统中常常包含隐式的输入,所以我们很难直接获取用户的需求。(如同 Watson 系统所做的工作一样[8],有时我们可以通过其他的应用程序从用户的交互行为中提取隐式需求。)

解决偏好不稳定因素常常需要基于知识的推荐技术。研究用户过去的交互行为被证明是一种妨害而不是帮助。然而，一些特殊的情况，如网站个性化推荐系统，用户在一个会话期间可以提供大量的隐含数据。这可以使我们能够组织成有效的文件信息与其他用户的信息进行对比。

表 11.2 展示了不同领域因素在推荐方法选择上的影响。

现在针对具体评判选择技术的标准还没有一个明确的结论，因为需要我们注意的一点是不同的技术有不同的实现方法和不同的维护成本。协同过滤推荐似乎是一种最小的开销实现方法。协同过滤推荐仅仅需要一个用户评分数据组成的数据库，并且数据库中的数据也不需要很干净，物品的特征也不需要精心设计。与其他推荐技术相比，协同过滤推荐的要求最少。基于知识的推荐系统是开销最大的一种方法，因为它需要特定的知识工程和持续维护。所以，开发者希望采用与领域相结合的一种开销最小的解决方案。

表 11.2 领域因素和推荐技术

因素		协同过滤	基于内容	基于领域知识
异质	低	✓	✓	✓
	高	+	—	—
风险	低	✓	✓	✓
	高	—	—	+
变动	低	✓	✓	✓
	高	—	+	+
交互方式	隐式	+	+	—
	显式	✓	✓	+
偏好	稳定	✓	✓	✓
	不稳定	—	—	+
解释	需要	—	—	+
	不需要	✓	✓	✓

还有另外一种考虑就是采用混合推荐，它融合了多种推荐技术。举例来说，为了解决混杂数据环境下偏好不稳定的问题，我们可以采用混合了基于内容的推荐和协同过滤推荐的方法。具体内容请参看文献[11]。

### 11.6.1 算法

如果已经能够明确地知道以什么样的特征为标识的领域适合这样的推荐技术或另一种推荐技术，那么接下来很自然的一个问题就是什么样的算法适合推荐。完全讨论这个问题已经超过了本章的内容范围，读者需要回顾相关的章节。但是就协同过滤推荐而言我们还是可以提出一些想法的。

在基于用户的协同过滤推荐中，根据活跃用户相似度高的原则选择合适的用户群。在系统运行过程中，使用加权的评分集合预测活跃用户的评分状况。不同的协同过滤实现方式使用不同的改进过的基于领域的评分预测算法。Herlocker 等在其文章<sup>[25]</sup>中提出了一种在这类算法中关于如何设计选择的实验分析，并分析了相似度矩阵、加权方法、合并方法和评分标准化的差异。

基于物品的协同过滤推荐是一种基于内存的算法，这个算法研究了物品之间的关系以及用户是如何评价它们的。基于物品的 kNN 算法被证明与基于用户的协同过滤算法相比可以产生更好更准确的推荐，尤其是在大的物品集合中[59]。

基于内存的最近邻居算法有两个重要的计算约束：规模和稀疏性。由于进行平方比较时需要比较每个用户和其他用户的区别，这些技术在大用户集合中显得没有多大用处。同样如果需要直接进行比较物品的评分，这意味着如果在一个高度稀疏的环境下用户将有很少的邻居。

在一些数据库中，过度稀疏的数据空间可能会隐藏稠密数据空间下的真相。热门度指数曲线使得采用基于内存的技术成为可能，因为它可以发现在数据稠密子空间下用户或物品之间的一致性，并且使用这个一致性在稀疏空间对用户进行推荐[27](Jester 在他的文

章中[20]提出构建一个高度稠密的数据空间给所有用户推荐)。

降维(降维的方法主要有:奇异值分解、隐语义分析或者其他的一些技术)是到目前为止解决评分数据稀疏的一种标准方式[60, 74]。各种形式的压缩及降维都需要严格的离线计算。通过这样的一个过程数据规模就得到了优化。由 Netflix prize 提出的电影评价分据也被用在 2007 年 KDD 杯数据挖掘大赛上, 这个电影数据集就是一个经典的高度稀疏的大数据集。这激发着越来越多的研究团队研究新的基于模型的算法[56, 33]。

这里还有其他的基于模型的协同过滤算法, 比如, 一些机器学习技术像贝叶斯网[7]和聚类分析[7, 66]。贝叶斯网络适用于用户兴趣比较稳定的领域, 因为用户兴趣变化缓慢就给构建模型提供了足够的时间。

聚类分析算法依据用户兴趣将用户分成不同的组。一旦建立好了一个簇, 那么对于个体的预测就可以根据簇中群体的平均值给出[59]。聚类技术产生的推荐与其他技术相比缺少人性化, 并且在有些时候聚类的结果要低于近邻算法[71]。聚类分析技术可以用在消减备选用户数据集的第一步, 也可以将最近邻计算的结果分散在不同的推荐引擎中。然而, 将用户分到不同的簇中降低了推荐的精确度, 但是预聚类是一种可以权衡准确度和流行度的方法[59]。

11.6.2    抽样推荐领域

表 11.3 阐述了这些评判标准在 10 个不同推荐领域中的应用。并非上面讨论过的 6 个方面都涉及, 但是我们也可以清楚地看到上面的方法还是相对准确的。高风险领域导致产生了基于知识的推荐; 可解读推荐也是一种好的预测器。异构领域主要依靠协同过滤推荐。当我们以高变动和偏好不稳定的视角去看待网页推荐的时候会有一点对立, 因为网页推荐主要依靠协同过滤方法。正如我们上面讨论过的, 数据库的大小会补偿偏好不稳定。此外推荐系统可以在一个会话内搜集大量明确的用户偏好数据。当然, 如果异构性很高, 那么我们往往偏向使用社群知识。

表 11.3    推荐系统的抽样领域

领域	风险	动力	多样性	性能	互动类型	可解读性	案例	技术
新闻	低	高	低	稳定?	不明显	不需要	Yahoo 新闻[6]、ACR 新闻[45]和 Google 新闻[16]	基于内容协同过滤
电子商务	低	高	高	稳定	不明显	不需要	Amazon、eBay	协同过滤
网页推荐系统	低	高	高	不稳定	不明显	不需要	[9, 36, 4]	协同过滤混合
电影	低	低	低	稳定	不明显	不需要	NetFlix[50, 64]、Movielens[21]	协同过滤
音乐	低	低	低	稳定?	不明显	不需要	Pandora 和[24, 28, 14]	协同过滤混合
财经服务 人寿保险	高	低	低	稳定	不明显	需要	Koba4Ms[17]、FSAdvisor[19][65]	基于知识
软件工程	低	低	低	稳定	明显	需要	[13]和[29]	混合和基于内容
旅游业	高	低	低	不稳定	明显/不明显	需要	旅行推荐系统[55][37]	基于内容基于知识
职位搜索	高	低	低	稳定	明显	需要	CASPER[35]和[39]	基于内容
地产	高	低	低	稳定	明显	需要	RentMe[10]、FlatFinder[67]和[73]	基于知识

## 11.7 总结

本章主要论述了推荐系统作为智能系统所需要的知识源。理解推荐方法的不同的最好方法是理解推荐算法所采用的知识源。通过考虑领域特征是如何影响知识源的可用性和质量的, 就可以将推荐技术和领域特征联系起来。

我们已经测试过 6 个不同的因素: 异构性、风险性、变动性、偏好稳定性、交互风格和可理解性。这些因素可以用于描述不同领域的特征并且使特征和推荐技术对应。这些评价方式在某些现有系统上的应用表明, 它们是在研究和应用领域得到成功采用的技术的晴雨表。

## 致谢

这篇论文基于 Ramezani 女士在 2007 年夏天于 IBM Watson 研究中心的工作。此外, 一些作者如 Lawrence Bergman、Rich Thompson 和 Bamshad Mobasher 等修改简化了本文的一些内容。期间 Bamshad Mobasher 在 2008 年的推荐系统与协同过滤智能用户交互大会上发表了题为“Selecting and Applying Recommendation Technologies”的报告。

## 参考文献

1. Aciar, S., Zhang, D., Simoff, S., Debenham, J.: Informed recommender agent: Utilizing consumer product reviews through text mining. In: WI-IATW '06: Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology, pp. 37–40. IEEE Computer Society, Hong Kong (2006).
2. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.* **23**(1), 103–145 (2005).
3. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* **17**(6), 734–749 (2005).
4. Anand, S.S., Mobasher, B.: Introduction to intelligent techniques for web personalization. *ACM Trans. Internet Technol.* **7**(4), 18 (2007).
5. Berkovsky, S., Aroyo, L., Heckmann, D., Houben, G.J., Kröner, A., Kuflik, T., Ricci, F.: Providing context-aware personalization through cross-context reasoning of user modeling data. In: S. Berkovsky, K. Cheverst, P. Dolog, D. Heckmann, T. Kuflik, P. Mylonas, J. Picault, J. Vassileva (eds.) *UbiDeUM'2007 - International Workshop on Ubiquitous and Decentralized User Modeling*, at User Modeling 2007, 11th International Conference, UM 2007, Corfu, Greece, June 26, 2007, Proceedings (2007).
6. Billsus, D., Pazzani, M.J.: User modeling for adaptive news access. *User Modeling and User-Adapted Interaction* **10**(2-3), 147–180 (2000).
7. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43–52 (1998).
8. Budzik, J., Hammond, K., Birnbaum, L.: Information access in context. *Knowledge based systems* **14**(1-2), 37–53 (2001).
9. Buono, P., Costabile, M.F., Guida, T., Piccinno, A.: Integrating user data and collaborative filtering in a web recommendation system. In: *Lecture Notes in Computer Science: Hypermedia: Openness, Structural Awareness, and Adaptivity*, pp. 192–196. SpringerLink (2002).
10. Burke, R.: Knowledge-based recommender systems. In: *Encyclopedia of Library and Information Systems*. Marcel Dekker (2000).
11. Burke, R.: Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* **12**(4), 331–370 (2002).
12. Burke, R.: Interactive critiquing for catalog navigation in e-commerce. *Artif. Intell. Rev.* **18**(3-4), 245–267 (2002).

13. Castro-Herrera, C., Duan, C., Cleland-Huang, J., Mobasher, B.: Using data mining and recommender systems to facilitate large-scale, open, and inclusive requirements elicitation processes. In: RE '08: Proceedings of the 2008 16th IEEE International Requirements Engineering Conference, pp. 165–168. IEEE Computer Society, Washington, DC, USA (2008).
14. Chen, H.C., Chen, A.L.P.: A music recommendation system based on music data grouping and user interests. In: CIKM '01: Proceedings of the tenth international conference on Information and knowledge management, pp. 231–238. ACM (2001).
15. Chen, L., Pu, P.: Survey of preference elicitation methods. Tech. rep., Swiss Federal Institute of Technology in Lausanne (EPFL), Lausanne, Switzerland (2004).
16. Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization: scalable online collaborative filtering. In: WWW '07: Proceedings of the 16th international conference on World Wide Web, pp. 271–280. ACM, New York, NY, USA (2007).
17. Felfernig, A.: Koba4ms: Selling complex products and services using knowledge-based recommender technologies. In: CEC '05: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology (CEC'05), pp. 92–100. IEEE Computer Society (2005).
18. Felfernig, A., Burke, R.: Constraint-based recommender systems: technologies and research issues. In: ICEC '08: Proceedings of the 10th international conference on Electronic commerce, pp. 1–10. ACM, New York, NY, USA (2008).
19. Felfernig, A., Kiener, A.: Knowledge-based interactive selling of financial services with fsadvisor. In: Proceedings of the National Conference on Artificial Intelligence, pp. 1475–1482 (2005)
20. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. *Inf. Retr.* 4(2), 133–151 (2001).
21. Good, N., Schafer, J.B., Konstan, J.A., Borchers, A., Sarwar, B.M., Herlocker, J.L., Riedl, J.: Combining collaborative filtering with personal agents for better recommendations. In: AAAI:Conference on Artificial Intelligence, pp. 439–446 (1999).
22. Hauptmann, A.G.: Integrating and using large databases of text, images, video, and audio. *Intelligent Systems and Their Applications*, IEEE 14(5), 34 – 35 (1999)
23. Hayes, C., Avesani, P., Veeramachaneni, S.: An analysis of the use of tags in a blog recommender system. In: M.M. Veloso (ed.) IJCAI-07, the International Joint Conference on Artificial Intelligence, pp. 2772–2777 (2007)
24. Hayes, C., Cunningham, P.: Smart radio: Building music radio on the fly. In: Proceedings of Expert Systems 2000, Cambridge, UK (2000)
25. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 230–237. ACM Press, New York, NY, USA (1999).
26. Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. In: CSCW '00: Proceedings of the 2000 ACM conference on Computer supported cooperative work, pp. 241–250. ACM Press (2000).
27. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* 22(1), 5–53 (2004).
28. Hijikata, Y., Iwahama, K., Nishida, S.: Content-based music filtering system with editable user profile. In: SAC '06: Proceedings of the 2006 ACM symposium on Applied computing, pp. 1050–1057. ACM (2006).
29. Holmes, R., Walker, R.J., Murphy, G.C.: Approximate structural context matching: An approach to recommend relevant examples. *IEEE Transactions on Software Engineering* 32(12), 952–970 (2006).
30. Kosala, R., Blockeel, H.: Web mining research: a survey. *SIGKDD Explor. Newsl.* 2(1), 1–15 (2000).
31. Kovacs, A.I., Ueno, H.: Recommending in context: A spreading activation model that is independent of the type of recommender system and its contents. In: G. Uchyigit (ed.) Proceedings of Workshop on Web Personalisation, Recommender Systems and Intelligent User Interfaces In conjunction with AH 2006:International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems. Dublin, Ireland (2006)
32. Krulwich, B.: Lifestyle finder: Intelligent user profiling using large-scale demographic data. *Artificial Intelligence Magazine* 18(2) (1997)
33. Kurucz, M., Benczúr, A.A., Kiss, T., Nagy, I., Szabó, A., Torma, B.: Kdd cup 2007 task 1 winner report. *SIGKDD Explor. Newsl.* 9(2), 53–56 (2007).
34. Lang, K.: Newsweeder: Learning to filter netnews. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 331–339 (1995)



35. Lee, D.H., Brusilovsky, P.: Fighting information overflow with personalized comprehensive information access: A proactive job recommender. In: ICAS '07: Proceedings of the Third International Conference on Autonomic and Autonomous Systems, p. 21. IEEE Computer Society, Washington, DC, USA (2007).
36. Li, J., Zaiane, O.R.: Combining usage, content, and structure data to improve web site recommendation. *Lecture Notes in Computer Science : E-Commerce and Web Technologies* pp. 305–315 (2004)
37. Mahmood, T., Ricci, F., Venturini, A., Hpken, W.: Adaptive recommender systems for travel planning. In: P. O'Connor, W. Hpken, U. Gretzel (eds.) *Information and Communication Technologies in Tourism 2008*, pp. 1–11. Springer (2008).
38. Maidel, V., Shoval, P., Shapira, B., Taieb-Maimon, M.: Evaluation of an ontology-content based filtering method for a personalized newspaper. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 91–98. ACM (2008).
39. Malinowski, J., Keim, T., Wendt, O., Weitzel, T.: Matching people and jobs: A bilateral recommendation approach. In: *HICSS '06: Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, pp. 137c–137c. IEEE Computer Society (2006).
40. Markines, B., Stoilova, L., Menczer, F.: Bookmark hierarchies and collaborative recommendation. In: *Proceedings of the Twenty-First AAAI Conference on Artificial Intelligence*. AAAI Press (2006)
41. McSherry, D.: Explaining the pros and cons of conclusions in cbr. In: P.A.G. Calero, P. Funk (eds.) *Proceedings of the European Conference on Case-Based Reasoning (ECCBR-04)*, pp. 317–330. Springer (2004). Madrid, Spain
42. McSherry, D.: Explanation in recommender systems. *Artif. Intell. Rev.* **24**(2), 179–197 (2005)
43. McSherry, D., Aha, D.W.: Mixed-initiative relaxation of constraints in critiquing dialogues. In: *ICCBR '07: Proceedings of the 7th international conference on Case-Based Reasoning*, vol. 4626, pp. 107–121 (2007)
44. Mladenic, D., Grobelnik, M.: Feature selection for unbalanced class distribution and naive bayes. In: *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, pp. 258–267. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999)
45. Mobasher, B., Cooley, R., Srivastava, J.: Automatic personalization based on web usage mining. *Communications of the ACM* **43**(8), 142–151 (2000).
46. Montaner, M., López, B., Rosa, J.L.D.L.: A taxonomy of recommender agents on the internet. *Artif. Intell. Rev.* **19**(4), 285–330 (2003).
47. Mooney, R.J., Roy, L.: Content-based book recommending using learning for text categorization. In: *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, pp. 195–204. ACM Press (2000).
48. Moskovitch, R., Elovici Y., Rokach L., Detection of unknown computer worms based on behavioral classification of the host, *Computational Statistics and Data Analysis*, **52**(9):4544–4566 (2008)
49. Niwa, S., Doi, T., Honiden, S.: Web page recommender system based on folksonomy mining for itng 06. In: *Information Technology: New Generations, 2006. ITNG 2006. Third International Conference on*, pp. 388–393 (2006).
50. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In: *Proc. of the of the KDD Cup and Workshop 2007 (KDD 2007)* (2007)
51. Pazzani, M., Billsus, D.: Learning and revising user profiles: The identification of interesting web sites. *Machine Learning: Special issue on multistrategy learning* **27**(3), 313–331 (1997).
52. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. *Artif. Intell. Rev.* **13**(5–6), 393–408 (1999)
53. Rack, C., Arbanowski, S., Steglich, S.: A Generic Multipurpose recommender System for Contextual Recommendations, pp. 445–450. IEEE Computer Society, Washington, DC, USA (2007).
54. Resnick, P., Varian, H.R.: Recommender systems. *Commun. ACM* **40**(3), 56–58 (1997)
55. Ricci, F.: Travel recommender systems. In: *IEEE Intelligent Systems*, pp. 55–57 (2002)
56. Rosset, S., Perlich, C., Liu, Y.: Making the most of your data: Kdd cup 2007 "how many ratings" winner's report. *SIGKDD Explor. Newsl.* **9**(2), 66–69 (2007).
57. Roth-Berghofer, T.R.: Explanations and case-based reasoning: Foundational issues. In: *Advances in Case-Based Reasoning*, pp. 389–403. Springer Verlag (2004)
58. Salam, M., Reilly, J., McGinty, L., Smyth, B.: Knowledge discovery from user preferences in conversational recommendation. In: *Knowledge Discovery in Databases: PKDD 2005*, pp. 228–239. Springer Berlin / Heidelberg (2005)
59. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommen-

- ation algorithms. In: WWW '01: Proceedings of the 10th international conference on World Wide Web, pp. 285–295. ACM (2001).
60. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of dimensionality reduction in recommender systems—a case study. In: Proceedings of ACM WebKDD Workshop (2000)
61. Schafer, J.B., Konstan, J.A., Riedl, J.: E-commerce recommendation applications. *Data Mining and Knowledge Discovery* 5(1-2), 115–153 (2001)
62. van Setten, M., Pokraev, S., Koolwaaij, J.: Context-aware recommendations in the mobile tourist application compass. In: W. Nejdl, P. De Bra (eds.) *Adaptive Hypermedia 2004*, pp. 235–244. Springer Verlag (2004).
63. Smyth, B., Balfe, E., Freyne, J., Briggs, P., Coyle, M., Boydell, O.: Exploiting query repetition and regularity in an adaptive community-based web search engine. *User Modeling and User-Adapted Interaction* 14(5), 383–423 (2005).
64. Takács, G., Pilászy, I., Németh, B., Tikk, D.: On the gravity recommendation system. In: Proc. of the of the KDD Cup and Workshop 2007 (KDD 2007), pp. 22–30 (2007)
65. Tartakovski, A., Schaaf, M., Bergmann, R.: Retrieval and configuration of life insurance policies. In: *Lecture Notes in Computer Science, Case-Based Reasoning Research and Development*, pp. 552–565. Springer Berlin / Heidelberg (2005)
66. Ungar, L., Foster, D., Andre, E., Wars, S., Wars, F.S., Wars, D.S., Whispers, J.H.: Clustering methods for collaborative filtering. In: *Workshop on Recommender Systems at the 15th National Conference on Artificial Intelligence*. AAAI Press (1998)
67. Viappiani, P., Pu, P., Faltings, B.: Conversational recommenders with adaptive suggestions. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 89–96. ACM (2007).
68. Wu, H., Zubair, M., Maly, K.: Harvesting social knowledge from folksonomies. In: *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*, pp. 111–114. ACM Press (2006).
69. Xu, Z., Fu, Y., Mao, J., Su, D.: Towards the semantic web: Collaborative tag suggestions. In: *Proceedings of the Collaborative Web Tagging Workshop at the WWW 2006*. Edinburgh, Scotland (2006)
70. Yu, Z., Zhou, X., Zhang, D., Chin, C.Y., Wang, X., Men, J.: Supporting context-aware media recommendations for smart phones. *Pervasive Computing* 5(3), 68–75 (2006).
71. Zaiane, O.R., Han, J., Li, Z.N., Chee, S.H., Chiang, J.Y.: Multimediaminer: a system prototype for multimedia data mining. In: *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pp. 581–583. ACM (1998).
72. Zanker, M.: A collaborative constraint-based meta-level recommender. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 139–146. ACM, New York, NY, USA (2008).
73. Zhang, J., Pu, P.: A comparative study of compound critique generation in conversational recommender systems. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4018 NCS, pp. 234–243. Springer Verlag, Heidelberg, D-69121, Germany (2006)
74. Zhang, S., Wang, W., Ford, J., Makedon, F., Pearlman, J.: Using singular value decomposition approximation for collaborative filtering. In: *CEC '05: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology*, pp. 257–264. IEEE Computer Society, Washington, DC, USA (2005).

# 用于技术强化学习的推荐系统

Nikos Manouselis、Hendrik Drachsler、Riina Vuorikari、Hans Hummel 和 Rob Koper

**摘要** 技术强化学习 (Technology Enhanced Learning, TEL) 旨在设计、开发和测试那些支持并加强个人和组织学习实践的社会技术创新。因此, 该应用领域包括了一般情况下的各种形式的教学和学习活动的技术支持。由于信息检索(搜索相关学习资源帮助老师或学生)在技术强化学习中至关重要, 而推荐系统的部署则大大提升了这一过程的趣味性。本章将着重介绍用于技术强化学习环境的推荐系统, 同时, 也将突出其不同于其他应用领域的推荐系统的一些特性。

## 12.1 简介

技术强化学习旨在设计、开发和测试那些支持并加强个人与组织学习实践的社会技术创新。因此, 该应用领域包括一般情况下各种形式的教学和学习活动的技术支持。由于信息检索(搜索相关学习资源帮助老师或学生)在技术强化学习中至关重要, 而推荐系统的部署则大大提升了这一过程的趣味性。

在任何产品多样性急剧增长的领域, 技术强化学习都需要做到能够更好地找到学习资源(主要是数字化资源)。例如, 在过去的几年间里, 人们建立了众多数字化学习资源库[96]。在美国, 突出的例子是资源库, 例如 MERLOT(<http://www.merlot.org>)有着超过 20 000 种学习资源(和大约 70 000 位注册用户), 以及 OER Commons([www.oercommons.org](http://www.oercommons.org))有着大约 18 000 种资源。在欧洲, 比较典型的是“欧洲校园网内的学习资源互换”(<http://lreforschools.eun.org>), 该网站含有超过 43 000 种学习资源, 这些资源来自欧洲和其他地区 25 个不同的内容提供商。除学习内容, 学习资源可能还包括学习路径(帮助他们寻找适合的学习资源)或相关的学习同伴(与其一起学习)。

在众多的网上学习资源可供选择的情况下, 考虑到各种正式和非正式环境下与这些资源互动的机会, 技术强化学习系统中的所有用户群将会很快从这些庞大的学习资源中确定适合他们的学习资源。因此, 技术强化学习本身就包含推荐系统的概念。最近有关综合该领域相关研究成果, 并推动研究人员关注相似课题的努力是“用于技术强化学习

---

Nikos Manouselis, Greek Research and Technology Network (GRNET S. A.), 56 Messogeion Av., 115 27, Athens, Greece e-mail: [nikosm@ieee.org](mailto:nikosm@ieee.org)

Hendrik Drachsler, Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit Nederland e-mail: [hendrik.drachsler@ou.nl](mailto:hendrik.drachsler@ou.nl)

Riina Vuorikari, European Schoolnet (EUN), 24, Rue Paul Emile Janson, 1050 Brussels, Belgium e-mail: [riina.vuorikari@eun.org](mailto:riina.vuorikari@eun.org)

Hans Hummel, Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit Nederland e-mail: [hans.hummel@ou.nl](mailto:hans.hummel@ou.nl)

Rob Koper, Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit Nederland e-mail: [rob.koper@ou.nl](mailto:rob.koper@ou.nl)

翻译: 谢冰莹, 余亮, 郑州大学-吴宾 审核: 吉林大学-丁彬钊, 北京大学-王雪丽

的社会信息检索(SIRTEL)”的年度研讨会系列,并作为一个关于“用于技术强化学习的社会信息检索”特殊专刊发行在《数字信息期刊》上。这些努力带来了一系列有趣的结果,主要包括:

- 大量的推荐系统已被部署(或正在部署)在技术强化学习环境中;
- 技术强化学习推荐系统要尝试达到的信息检索目标不同于用于其他系统(如产品推荐系统)的推荐系统;
- 需要确定技术强化学习推荐系统的特性,以便详细说明其用于系统设计、开发和评估的方法。

在此方向内,本章试图介绍用于技术强化学习环境的推荐系统部署的相关议题,请牢记该应用领域的主要特性。本章主要贡献如下:

- 讨论推荐系统在 TEL 的应用背景,特别是在 TEL 语境下的相关特性。
- 考虑 TEL 环境中的用户任务支持,以及如何将他们同用于其他推荐系统内较典型的用户任务比较。
- 回顾来自自适应教育超媒体系统和学习网络(LN)概念的相关工作。
- 评估当前的 TEL 推荐系统开发状态。
- 提供关于 TEL 推荐系统大概的特性和要求介绍,为教育系统的进一步应用和研究提供基础。

## 12.2 背景

### 12.2.1 TEL 作为上下文

TEL 涉及教育环境中产生的不同类型的数据,这通常被称为宏观环境[99]。这个概念对用户动作可能具有重大影响力,这些宏观环境维度下的例子包括规模(如教育水平),正式和非正式的学习,投放环境,不同的用户角色。教育水平的例子是 K-12 教育、高等教育(HE)、职业教育和培训(VET)。

学习的正式环境,包括教育机构(如大学、学校)提供的学习课程或课程框架,具有高度结构化的特点,通向一个特定的认证和领域专家来保证质量。这通常发生在老师指导的环境下,在现场、同步方式下实现人机互动。

非正式的环境,换句话说,即在文献中描述所谓的终身学习者,没有参加任何正式的学习作为一个学习阶段,负责自己的学习节奏和路径[17, 64]。这个学习过程很大程度上依赖个人的偏好或者通常选择自我引导[8]。非正式学习的资源可能来自专家社区、工作环境、培训、甚至朋友可能会提供的机会和非正式能力的培养。

参与 TEL 的特点是可以提供混合学习机会以实现纯粹的远程教育[71]。混合式学习结合了传统的面对面的学习与以电脑辅助教学的学习[36]。远程教育,换句话说,能在任意的同步或异步方法下交付使用 TEL 环境。传统上,远程教育更接近于自学和与学习材料互动,典型的是使用异步的方式进行[36]。但是,实时媒体流和虚拟个人学习环境已经在正式的教育环境中促进同步远程学习服务的发展。

最后,在 TEL 中可以找到不同的角色和需求。老师指导的互动和学习者指导的学习过程之间是有区别的。这会对 TEL 环境的目标用户产生影响。尽管宏观环境对解释和设计的影响巨大,但方方面面很容易达成一致,而且相对容易衡量。微环境则是一个比较有争议的概念,更难以衡量。但是宏观环境一般是领域特有的,而微环境的概念在各种不

同领域都存在。

12.2.2 TEL 推荐的目标

推荐系统过去的成果中包括许多相关的用户任务，这使得推荐系统能够支持在一些特定的上下文相关的应用场景(见第 7 章)。更具体地说，Herlocker 等在文献[38]中将热门(或不那么热门)的用户任务与表 12.1 中包括的许多特定的推荐目标联系起来。一般来说，大多数这些已经确定的目标和任务在 TEL 推荐系统中也是有效的。例如，在支持学生达到特定学习目标的推荐系统中，“在文档中提供注释”或者“推荐学习资源的序列”都是相关的任务。表 12.1 给出了推荐结果如何支持 Herlocker 等在文献[38]中发现所有任务的 TEL 相关活动。此外，它还包括有关任何额外需求的点评，这将有助于 TEL 推荐系统的开发者。

表 12.1 当前推荐系统支持的用户任务和 TEL 推荐系统的需求

	任务	描述	通用推荐系统	TEL 推荐系统	新需求
推荐系统支持的现有用户任务	1. 上下文注释	当用户执行其他任务时的推荐结果	比如，预测网页中的链接有多相关	比如，预测课程阅读列表中的物品的相关性或有有用性	发现能在学习上下文中表示相关性和有用性的属性
	2. 发现好的物品	建议物品的推荐	比如，收到要访问网页的列表	比如，收到有关某个主题的在线教育资源的列表	—
	3. 发现所有好的物品	所有相关物品的推荐	比如，收到有关某个主题的参考文献的完整列表	比如，建议有关某个主题的科学文献和博文的完整列表	—
	4. 推荐序列	一系列物品的推荐	比如，收到推荐的一系列歌曲	比如，收到从资源到达成特定学习目的的推荐序列	发现能表示特定学习目的的相关性的正式和非正式的属性
	5. 仅仅浏览	当用户在浏览时意外推荐	比如，买了这个的用户还买了什么	比如，在大学网站上收到新课程的推荐	发现能在学习上下文中表示相关性和有用性的正式和非正式的属性
	6. 发现可信推荐	在最初探索和测试系统阶段时的推荐	比如，你肯定会喜欢的电影	比如，将课程推荐限制在非常信任的范围内	发现能在正式和非正式学习中衡量可信度和可靠性的标准
推荐系统可能支持的 TEL 用户任务	1. 发现新奇的资源	推荐特别新或新奇的物品	比如，收到最近新增或特别有争议物品的推荐	比如，收到所覆盖主题上非常新和有争议的资源	发现能超越相似度选择物品的推荐技术
	2. 发现同伴	推荐有相关兴趣的其他人	比如，被建议有着相似兴趣的用户个人信息	比如，被建议同班同学中的同伴	发现能衡量其他人相似度的属性
	3. 发现好的路径	在学习资源中推荐替代的学习路线	比如，收到相似歌曲的替代系列	比如，收到未达到某个特定学习目的，在相同资源上的替代学习路线	发现构建和建议替代(但相似)序列的标准

另一方面，与典型的物品推荐方案比较，有几个关于期望什么学习类型的特殊细节需要考虑，比如，学习新的知识或者加强现存知识可能需要不同类型的学习资料。这反映在表 12.1 中的第二部分，带有用户任务的例子对于 TEL 特别值得关注。同样，这里也包括了针对 TEL 推荐系统开发者的任意额外需求的点评。

除了初始任务识别，在 TEL 上下文环境中的推荐有许多基于丰富教学理论和模型的细节。例如，对于没有特定领域先验知识的学习者，可以应用如 Vygotsky 的“最近发展



区”的相关教学规则,例如,“推荐学习对象应该有一个水平略高于学习者现有的能力水平”[102]。不同于购买产品,学习是一种努力,相比于商业交易往往需要更多的时间以及相互交流。在达到固定时间后学习者很少达到最终状态。不同于购买以及拥有产品,学习者会获得在不同的领域不同层次的不同能力。在这种情况下,确定相关学习目标并且支持学习者获得目标是重要的。另一方面,根据上下文,一些特定的用户任务可优先。这就要求推荐的时间跨度要长于推荐产品之一或者是相似学习资料的推荐产品,因为学习过程中心任务是概述和重复[68]。

对于以教师为中心的教学环境,需要支持不同的任务。这些任务包括备课相关的任务,上一节课(也就是实际教学),以及评估或考核相关的任务。例如,要备好课教师会有一定的教育目标,并且需要保证授课方法适合学习者的特征(例如,他们以前的知识)。备课可以包括各种信息寻求任务,如寻找内容来调动学习者积极性,来唤醒现有知识,来阐述、具体化和描述新概念和信息。可以使用不同的教学方法来授课(无论是否支持 TEL),其有效性是根据设定的目标来评估。TEL 推荐系统可以支持一个或多个任务,产生多种推荐目标。

因此,尽管先前确定的用户任务和推荐的目标可以在一个 TEL 中被认为是有效的,但是仍有一些特殊性和复杂性。这意味着,将一个推荐系统,从现有的(如商业)内容推荐系统转化到 TEL 可能无法准确地满足目标用户的需求。在 TEL 中,定义推荐目标以及部署推荐系统之前,应该仔细分析目标用户和他们支持的任务是否应该被执行。这意味着 TEL 推荐的目标可以是相当复杂的:例如,一个典型的 TEL 推荐系统可以在各种学习资源中建议一些可供选择的学习路径,不管是以顺序学习序列的形式,还是以与学习资源交互的体系结构的形式。这在教育学上是一件有意义的事情,反映了基于用户精通程度、特定兴趣和意图应用环境的个人学习目的和目标能力等级。上下文一些环境变量必须考虑,如用户属性、域的特点以及提供个性化推荐的智能方法。在自适应教育超媒体系统领域,已经开展了大量有关这些主题的工作。

## 12.3 相关工作

Web 系统往往难以满足大量用户的异质性需求,针对这一难题,出现了一个新的研究分支叫作**自适应网络系统(或自适应超媒体)**,尝试让 Web 行为与目标用户的动机、任务、兴趣及其他用户感兴趣的特征相适应,来克服传统中“一刀切”做法中存在的弊端[12]。自适应系统的一个分类已经应用到教育领域,称作“**自适应教育超媒体(AEH)系统**”。由于 AEH 系统解决了与 TEL 推荐系统高度相关的一些问题,本节会对相关工作作简要概述,并指出其与 TEL 推荐系统之间的相同点与不同点。

### 12.3.1 自适应教育超媒体

自适应网络系统属于用户-自适应软件系统类别[87]。Oppermann[73]认为“可以根据用户需求自动改变自身特征”的系统可被认为是具有自适应性的。自适应系统会考虑用户与系统之间交互的方式,并相应地调整界面展示或系统行为[108]。Jameson[43]补充了一个重要特征:“用户自适应系统是一种交互系统,能基于对用户个体信息的显著推断,令行为具有自适性。”

自适应系统帮助用户从大规模的信息空间中寻找相关的物品。基本包含三种主要的自适应技术[12]:自适应内容选择、自适应导航支持、自适应呈现。其中第一个自适应内容

选择技术发源于自适应信息检索(IR)领域[6],并基于搜索来获取信息。当用户查询某些信息时,系统能自动选择和优先与之关联度最高的条目。自适应导航支持由自适应超媒体系统[9]引进,并基于浏览来获取信息。当用户由一个条目导向另一个条目时,系统能控制链接(如隐藏、分类、增加注释),从而将用户引导至关联度更高的信息条目上。自适应呈现技术来自智能系统中的自适应解释和自适应呈现研究[70,76]。它负责呈现信息而非信息获取。当用户访问一个特定页面时,系统能自适性地呈现其内容。

如 Brusilovsky[10]所描述的,教育超媒体是自适应系统最先应用的领域之一。图 12.1 展示的是一个自适应教育超媒体系统层级的简化结构。这一结构包含:教育内容(如学习资源)、领域(领域本体论)、用户(用户模型)等知识的呈现和组织层;适应机制和规则层;为用户提供运行时适应结果层。1990 年到 1996 年间一些自适性教育超媒体系统率先被开发出来,可以大致分为两种研究趋势。其中一种趋势的系统由智能教学系统(ITS)领域的研究者开创,尝试将该领域的传统学生建模和该领域中已开发的自适应方法拓展到具有超媒体成分的 ITS[14,34,77]。另一种趋势则由从事教育超媒体的研究者开创,试图使这一系统适用于学生个体[19,21,39,48]。AEH 研究通常遵循自顶向下的方法,主要依赖的是专业知识及参与,从而对 TEL 情境变量进行界定及模型化。例如,Cristea[18]描述了 AEH 内容授权时一些需要专业知识任务,首先要建立来源,对其进行标记,将其与已知领域模型结合;然后,建构和维护数据的或动态的用户模型,因为这是 AEH 成功自适的关键。总之,AEH 中大部分用户相关信息(特征化的需求及欲望)需要在内容生成阶段进行编码。这种适用于正式教育环境中,此时的情境变量往往是已知的,AEH 研究中有很多能在 TEL 推荐研究中用到(如处理学习者和领域模型)。另一方面,在非正式情境下,需要探索对专业知识需求较少的方法。

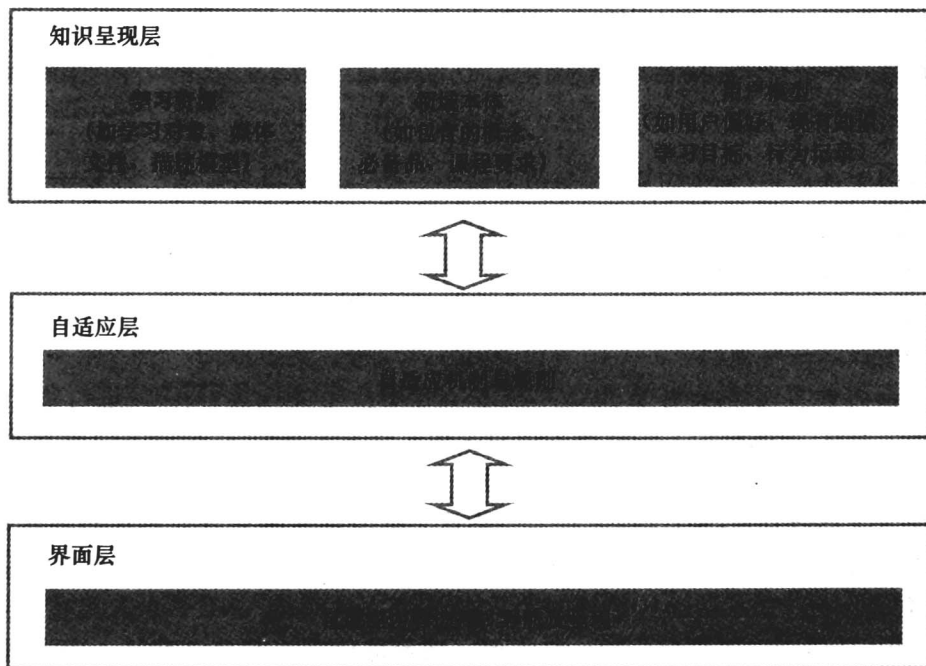


图 12.1 AEH 简化结构中包含的通用层(文献[47])

### 12.3.2 学习网络

另一研究方向中,语境变量可以从用户贡献的内容中提取。这一系统的分类包含学习

网络,能连接特定领域中分布的学习者和提供者[53, 54]。学习网络的设计与发展均是高度灵活、以学习者为中心、自底向上的,要胜过正式课程和自顶向下的以学习资源为中心的模型。学习网络由许多学习者和关联者提供的学习活动组成。每一个用户在任一时刻都可以添加、编辑、删除或评价学习资源。

学习网络的概念[54]为分散在各地的终身学习者提供了方法和技术设施来帮助他们提升个人能力(见图 12.2)。它利用了 Web 2.0 发展所带来的可能性,展现了网络化知识社会中学习的新趋势。学习网络是以学习者为中心的,其发展依赖学习者参与,是自底向上的结构。学习网络这一概念的核心思想是涌现(emergence),涌现在行为个体和资源的交互系统自行组织形成一个高层级行为模式的过程中出现[35, 45, 103]。

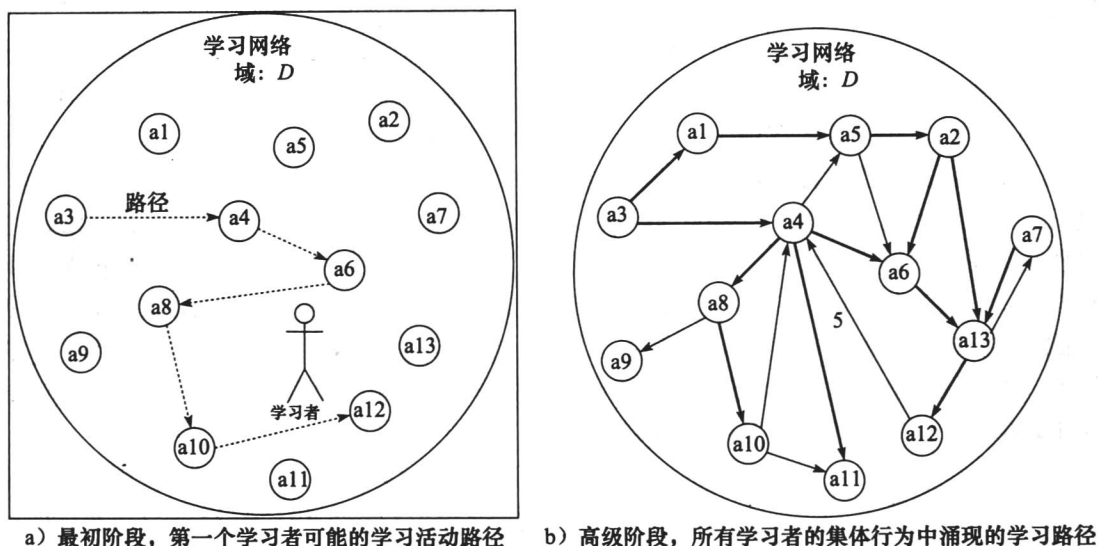


图 12.2 学习网络的进化

我们可以想象用户(如学习者)在学习网络中与学习活动进行交互,而这个过程是被记录下来的。通过间接测量,如时间及学习结果,或直接测量,如用户提供的评分和标签,可以确定一个学习网络中最易达成或者最受欢迎的路径(如[28, 100])。这些信息可以反馈到学习网络中的其他学习者那里,从而为网络中的“学习者群体”提供集体性的知识。大多数学习环境设计是自顶向下的,因为很多时候其结构、学习活动、学习流程由教育机构预先设定好,而学习网络采用 Web 2.0 技术,对用户生成内容(UGC)进行创造、分享、评分和调整。TEL 领域里,一些欧洲的项目采用了这些从底向上的知识创造与分享方法。欧盟倡议中的 TENcompetence 项目建立了这一非正式学习网络[110]。

凝聚型个人学习环境(MUPPLE)[109]是另一遵循自底向上方式形成和界定语境变量的系统类别,首先出现在[62, 63, 109]中。iCamp 欧盟倡议强调,通过营造灵活的氛围,允许学习者针对特定学习活动打造属于个人的环境,从而将 Web 2.0 来源整合进 MUPPLE。MUPPLE 是学习网络概念的一个实例,因而能体现出其部分特征。由于 MUPPLE 不需要学生有教育机构背景,专注的是学习者需求而非学生管理或者评估机构的需求,所以也能支持非正式学习。学习者不需要修读正式的课程,也不会得到关于能力水平的任何证明文件。MUPPLE 的一个通病在于,由短时框架来收集的数据量,及其非结构化的收集方式,这会导致用户和领域建模难度很高,同时也不够结构化。另一方面看,这也是推荐系统中探索用户和物品交互时(如确定用户和物品相似度时)的常见问题。

### 12.3.3 相同点与不同点

许多自适应教育超媒体(AEH)系统属于正式学习(如[3, 20, 57]),知识域呈分散粒状,因而能为学习者提供个性化的推荐。这些系统采用了类似元数据和本体等的技术[16]来对关系、条件、学习资源以及学习者模型的依赖度进行定义。它们大多用在封闭语料库程序上,教育设计者可以用语义关系对这些程序上的学习资源进行描述,因而是一种正式的学习服务。如前所述,在正规的教育场景中(如大学),都会有结构良好的正式关系,比如,预先设置好对应的地点、学生/老师资料,以及检验程序的学习计划(课程表)。所有这些元数据都可以用于课程推荐,或者通过对学生进行学习资源或学习环境的适配来达到个性化学习的目的[5]。该领域一个有意思的研究方向是自适应排序,即在对学习资源进行排序时,会考虑个体特征和偏好等因素[47]。AEH中,在运行时及学习系统维护时,存在大量的设计活动。此外,学习环境中的领域需要进行详细描述。这些方面,使得自适应排序和其他自适应超媒体在TEL推荐中对技术要求非常高。

当没有呈现出高度结构化的域模型时会出现非正式的学习网络。为了创建用户或者领域模型,需要用到挖掘技术。比如,在非正式学习中,用户的先验知识是一个非常分散的参数,因为它完全依赖于用户提供的没有任何标准化的数据。在处理先验知识动态且分散的特性,以及填补知识域模型的缺失方面,类似隐性语义分析的概率统计技术具有很好的前景[97]。非正式学习中维护和结构的缺失,也称为“开放语料问题”。在给定了无限制数量的文档不能手动结构化,且不能由领域概念和社区元数据索引[16]时,即出现开放语料问题。因此,如协同过滤等自底向上的推荐技术更为合适,因其几乎不需要任何维护,而且可以通过社区中的突现行为提升推荐表现。Drachsler等[25]分析了在学习网络中,不同种类的协同过滤技术是如何帮助学习者的。根据其结论,为了给推荐系统提供合适的导航支持,我们必须对非正式学习中不同的环境条件,如维护缺失、学习对象分散等因素加以考虑。学习网络主要基于用户生成信息和交互来构建。

除了提到的非正式学习中先验知识这点不同,从环境条件中得到的数据集也存在差异。一般来说,与需要预测的评分数量相比,从推荐系统中得到的评分数量非常小。基于小规模打分的有效预测,对推荐系统非常重要,且能影响对特定推荐技术的选择。正式学习可以依赖多领域专家或者基于多重准则的学生的常规性评估(如教育质量、技术质量、易用性)[67],而在非正式学习中,这些评估程序却少之又少,且为非结构化。正式学习环境,如大学,经常整合评估流程来进行常规的质量评价,汇报给资助机构,在这种整合的评估流程之下,数据集会更加密集。总而言之,在非正式学习情境中,数据集的特点是由评分数量稀少而导致的“稀疏性问题”。非正式学习中运用多标准评分(见第24章)可以帮助克服这一问题,这些多标准评分对学习社区来说必须是合理的。社群能够在多个层面上给学习资源打分,如所要求的先验知识水平(从新手到专家)、学习资源的展示风格,甚至吸引力水平,因为让学生满意且有动力学习,是非正式学习中一个至关重要的标准。这些明确的评分程序还须辅以间接测量指标,如使用“学习资源的用户数量”、“学习资源所进行的调整数量”来评估学习资源的时效性。

因而,非正式学习与正式学习这样结构完善的领域在多方面存在着差异。非正式学习的推荐系统可能会没有官方机构维护,大多依赖于社区,也没有界定明确的元数据结构。另外,与正式学习自顶向下的设计且建立学习服务(封闭语料库)不同,非正式学习是从社区(开放语料库)中自底向上涌现的。因而,将推荐系统从正式场景转移并应用到非正式场

景下会很困难(反之亦然), 因为二者在用户任务和推荐目的上往往具有本质差异。

Drachsler 等[24]根据在 TEL 中适用性和有用性为推荐技术给出重要评价, 其简述如表 12.2。该表总结了每一种技术的优缺点, 以及在 TEL 推荐中设想的有用性。然而, 它的目的只是作为讨论的基础, 未来还会有对已有推荐方法和技术的更具体精确的调查。

表 12.2    Drachsler 等发明的推荐技术及其对 TEL 的有用性[24]

	名称	简介	优点	缺点	对 TEL 的有用性
协同过滤 (CF) 技术	1. 基于用户的 CF	对同一物品打分相似的用户很可能品味相近, 根据这一假设, 这项技术给用户推荐他没有见过但已由其他相似用户评过分的物品	<ul style="list-style-type: none"><li>• 无内容分析</li><li>• 领域无关</li><li>• 推荐质量随时间提升</li><li>• 自底向上方式</li><li>• 惊喜性</li></ul>	<ul style="list-style-type: none"><li>• 新用户问题</li><li>• 新物品问题</li><li>• 大众品味</li><li>• 可扩展性</li><li>• 稀缺性</li><li>• 冷启动问题</li></ul>	<ul style="list-style-type: none"><li>• 经验越多结果越好</li><li>• 将用户归类到群 (基于相似的打分情况)</li></ul>
	2. 基于物品的 CF	聚焦于物品, 假设得分相似的物品很可能本身就是相似的。推荐具有高度关联性的物品 (根据物品的打分情况)	<ul style="list-style-type: none"><li>• 无内容分析</li><li>• 领域无关</li><li>• 推荐质量随时间提升</li><li>• 自底向上方式</li><li>• 惊喜性</li></ul>	<ul style="list-style-type: none"><li>• 新物品问题</li><li>• 大众品味</li><li>• 可扩展性</li><li>• 冷启动问题</li></ul>	<ul style="list-style-type: none"><li>• 经验越多结果越好</li></ul>
	3. 定型或人口统计学 CF	具有相似属性的用户之间是匹配的, 因而推荐被相似用户喜欢的物品 (基于用户数据而非评分)	<ul style="list-style-type: none"><li>• 无冷启动问题</li><li>• 领域无关</li><li>• 惊喜性</li></ul>	<ul style="list-style-type: none"><li>• 信息获取</li><li>• 信息量不足</li><li>• 只有大众品味</li><li>• 获取元数据信息</li><li>• 维护本体</li></ul>	<ul style="list-style-type: none"><li>• 将用户归类到群</li><li>• 经验越多结果越好</li><li>• RS 初始阶段推荐</li></ul>
基于内容 (CB) 的技术	4. 案例推理	假定用户喜欢某一物品, 很可能也喜欢相似的物品。推荐新的类似物品	<ul style="list-style-type: none"><li>• 无内容分析</li><li>• 领域无关</li><li>• 推荐质量随时间提升</li></ul>	<ul style="list-style-type: none"><li>• 新用户问题</li><li>• 过于特化</li><li>• 稀缺性</li><li>• 冷启动问题</li></ul>	<ul style="list-style-type: none"><li>• 让学习者知晓学习目标</li><li>• 对混合型 RS 有用</li></ul>
	5. 基于属性的技术	根据物品属性与用户偏好的匹配来推荐。属性由对用户重要性加权	<ul style="list-style-type: none"><li>• 无冷启动问题</li><li>• 无新用户/新物品问题</li><li>• 对偏好改变敏感</li><li>• 包含与非物品无关的功能</li><li>• 能建立用户需求与物品间的映射</li></ul>	<ul style="list-style-type: none"><li>• 无需学习</li><li>• 只有存在分类时有效</li><li>• 需要本体建模和维护</li><li>• 过于特化</li></ul>	<ul style="list-style-type: none"><li>• 对混合型 RS 有用</li><li>• 初始阶段推荐</li></ul>

12.4    TEL 推荐系统调查

TEL 领域引入了若干推荐系统向用户提供学习资源。因线上发布的学习资源种类非常多样, 且辅导者和学习者合作带来益处, 这些系统在这一过程有可能起到重要的教育性作用[81, 82, 59]。接下来将回顾近期提出的一些方法, 并对它们的发展和评估状态进行点评, 见表 12.3。



表 12.3 文献中报告已实行的 TEL 推荐系统

系统	状态	评估关键	评估者
Altered Vista[80, 81, 82, 105]	全系统	界面、算法、系统使用	用户
RACOFI[2, 61]	原型	算法	系统设计者
QSAI[78, 79]	全系统	—	—
CYCLADES[4]	全系统	算法	系统设计者
CoFind[29, 30]	原型	系统使用	用户
学习目标排序[88]	原型	系统使用	用户
进化型在线学习系统[90, 91, 92, 93]	全系统	算法、系统使用	模拟用户、用户
ISIS-混合式个性化推荐系统[28]	原型	算法、系统使用	用户
多属性推荐服务[67]	原型	算法	系统设计者
学习目标推荐模型[95]	设计	—	—
RecoSearch[32]	设计	—	—
拟真环境[72]	全系统	算法	模拟用户
ReMashed[26, 27]	全系统	算法、系统使用	用户
CourseRank[55, 56]	全系统	系统使用	用户
CBR 推荐界面[33]	原型	—	—
APOSDLE 推荐服务[1]	原型	—	—
A2M 推荐系统[86]	原型	—	—
Moodle 推荐系统[44]	原型	算法、系统使用	用户
LRLS[41]	原型	系统使用、学习者表现	用户
RPL 推荐系统[49]	原型	系统使用	系统设计者、用户

最先为学习资源建立的协同过滤系统是 Altered Vista 系统[81, 82, 83]。这一研究的目的在于探索如何收集用户对学习资源的评估，然后促使他们对资源质量进行口头传播。该研究团队同时也对一些相关问题进行了探究，如界面设计[82]，建立无授权元数据来存储用户评价[81]，系统设计和系统所用评价框架[83]，以及使用系统来给社区成员推荐有趣的资源、推荐喜好和观念相近的人等预测试及实证研究[83, 104]。

另一用于学习资源推荐的系统是 RACOFI(规则应用协同过滤)设计者系统[2, 60, 61]。RACOFI 结合了两种推荐方法：基于用户评分的协同过滤引擎及参照规则引擎，其中参照规则引擎负责挖掘学习资源之间存在的关联，并将这些关联规则用于物品推荐。RACOFI 研究还无法评估推荐系统在教育方面的价值，也不输出用户对系统的评估结果。RACOFI 技术现已应用到一家商业网站 inDiscover(<http://indiscover.net>)的音乐唱片推荐上。此外，有研究者报告称在其系统中也采用了 RACOFI 的方法[32]。

Rafaeli 等创办了用于学习资源分享、评估和推荐的 QSIA(问题分享与交互作业)[78, 79]。这一系统目前被应用在线社区情境中，目的是对学习活动中社会性元素加以利用，促进合作和在线推荐，进一步形成学习者社区。Rafaeli 等没有建立一个典型的自动化推荐系统，而是选择主要为用户控制的推荐程序作为 QSIA 的基础。也就是说，用户可以自行决定是否对推荐建议来源(好友)进行控制，或者是否使用一个协同过滤服务。系统已经应用在一些如职员、助教、高中老师和学生之间的知识分享的学习情景中，但目前还没有披露其评估结果[78, 79]。

学习资源协同过滤的这一流派系统中，CYCLADES 系统[4]构建的环境让用户可以

搜索、获取并评价 OAI 储备库中的数字资源(OAI, 开放存档倡议, <http://www.open-archives.org>)。OAI 是几家电子内容提供商为了降低互通成本而达成的非正式协议, 因而, 这一系统可以推荐不同存档中的内容, 并可以通过一个开放的框架获取资源。CYCLADES 推荐评估测试由 60 个用户进行预测试获得, 主要测试几种协同过滤算法的表现。

相关的系统有 CoFind 原型[29, 30], 同样也使用网络免费数字资源, 但使用的是一个用大众分类(标签)方法来进行推荐的新方法。CoFind 的开发者认为基于偏好的预测并不适用于学习情境, 因而, 像大众分类这种由用户驱动自底向上的分类更重要。

Manouselis 等[67]采用了典型的基于邻域集的协同过滤算法来实现学习目标推荐。这一研究的创新之处在于, 其中包含的算法是多属性的, 允许推荐服务将用户对学习资源的多维度评分考虑进去。该研究一个有趣的结果是, 与使用同种算法的初始实验相比(如[65]), 相同的算法其表现却是变化的, 这种变化取决于测试的情境。例如, 相同算法在电子商务[65]和 TEL 对比研究[67]的结果, 会引起备选集合的不同算法选择。这可能暗示着在一个情境中(如电影推荐)表现良好的推荐算法在另一情境中(如 TEL)并不一定具有同样的表现, 这一领域还需要更深入的实验(见第 7 章)。

Shen 和 Shen[88]采用了另一种学习资源推荐方法, 他们建立的针对学习目标的推荐系统以排序规则为基础, 能为用户提供主题的本体概念方面的指导。在确定学习者之间能力差距后, 规则被解除, 从而向学习者推荐合适的资源。网络教育学院的学生会对其进行预测试, 反馈其对系统的看法。

Huang 等[41]引入了类似的排序系统。它使用 Markov 链式模型来计算已排序课程中学习目标的转换概率。这一模型采用熵值法来发现一条或者多条推荐的学习路径。预测试在一所大学实行, 一共有 150 名用户参与。

Tang 和 McCalla 提出了一个进化的在线学习系统——面向网上的新的学习资源, 其中采用了混合的推荐服务[89, 90, 91, 92, 93]。该系统主要用于大学学生及行业从业人员之间存储、分享研究论文及术语。根据资源的内容和技术方面来描述(标记)资源, 但同时学习者也需要通过打分来给予反馈。推荐系统实现包含聚类模块(使用数据聚类技术将兴趣相近的用户归类)和协同过滤模块(使用经典的协同过滤技术, 根据相似兴趣确定用户所属类别)。作者对几种技术进行研究来帮助提高系统表现, 如运用虚拟(仿真)用户等[93]。同时也通过真实用户对系统进行评估[94]。

Janssen 等[44]提出了一个相当简单的推荐系统, 这一系统不考虑任何学习者偏好或者用户个人信息。但他们做了一个大规模的包含控制组和实验组的实验, 发现与控制组相比, 实验组在有效性(完成学习目标的比例)上的正向影响在效率(完成学习所花费的时间)上并不显著。

Nadolski 等[72]为不同推荐算法组成的混合推荐系统创建了一个仿真环境, 目的是对比他们之间在非正式学习网络中对学习者的影响。他们基于轻量协同过滤的推荐策略对比了不同的成本集约本体, 从而设立了实验组模拟不同种推荐技术组合。Nadolski 等的文献[72]检测了哪种推荐策略对学习者的影响最为有效。研究表明, 轻型协同过滤推荐策略的推荐结果并不如本体推荐策略准确, 但考虑到非正式学习网络中缺乏维护等的一些背景, 这种策略还是有价值的。这一研究也证实了, 与没有推荐系统的情况相比, 推荐系统能让用户更有效、更快捷、更满意地达到目的。研究还显示, 包含评分机制的轻量协同过滤推荐技术可以作为维护密集、自顶向下本体推荐策略的一个良好的替代方案。

ISIS 系统采用了 Hummel 等[42]近期提出的一种混合的学习资源推荐方案。作者在 Koper[52]的模拟研究之上,提出结合了基于社交(使用其他学习者的数据)和信息(学习者个人信息和活动的元数据)的混合推荐系统,同时设计了一个针对真实用户的实验。Drachsler 等[28]最近报告了 ISIS 实验的结果,发现在四个月的运行时间后,系统对学习者的效率(完成学习所花费的时间)存在显著影响。这是呈现学习者用户个人信息和活动的的一个很好的范例,这一应用是学习领域里的潮流方向。

这一研究团队最近开发了 ReMashed 推荐系统[26, 27],关注的是非正式学习网络中的学习者。他们建立了一个混聚环境,组合了不同 Web 2.0 服务,如 Flickr、Delicious.com、Slideshare.com 等的用户源。同时再一次采用了混合式推荐系统,利用了不同 Web 2.0 来源的标签和打分数据。之所以采用已有的 Web 2.0 网站上的标签是为了解决推荐系统中常见的冷启动问题(见第 19 章)。ReMashed 的用户可以对系统中所有用户数据进行打分,这些评分被用到基于 Duine 预测引擎的经典协同过滤推荐上[98]。

类似的方法还有学习目标推荐模型(LORM),它同样采用混合推荐算法的方法,从多属性描述资源,但目前还没有报告称应用在实际的系统中[95]。

在 CourseRank 系统(<http://courserank.stanford.edu/CourseRank/main>)中也采用了混合推荐法,作为斯坦福学生的非官方选课指南。在这个系统中,推荐过程被视为从课程和学生信息的关系数据库进行查询的过程[55]。为此,需要界定一些数组操作,让系统为其用户提供灵活的推荐。该系统最早于 2007 年 9 月投入使用,引起了用户的极大兴趣,据报告 70% 的斯坦福学生都使用了这一系统[56]。

Tunis 虚拟大学(RPL 平台, <http://cours.uvt.rnu.tn/rpl/>)课程储备库的原型系统中也用到了这一混合方法。原型里包含的推荐引擎,运用了协同过滤和内容过滤的算法组合,使用的是用户行为日志记录和挖掘得到的数据。RPL 平台上的使用日志也用于这一目的,目前已经开展了初级评估实验[49]。

最后,还有一些系统或算法提议可被用于支持学习资源的推荐,包括许多半成品系统,如 Gomez-Albarran 和 Jimenez-Diaz[33]的案例推理推荐系统、为大型机构雇员提供服务的 APOSDLE EU 物品(<http://www.aposdle.tugraz.at>)知识分享环境中的情境推荐系统[1]、A2M 原型[86]等。也有相关研究关注多媒体学习资源推荐在手机、掌上电脑等移动设备的延伸[51]。

尽管有越来越多的系统应用到学习资源推荐,对其发展和评估的进一步探究则发现,在现实应用背景下的评估研究缺少系统性。如表 12.3 所示:

- 提出的大多数系统(20 个中有 12 个)仍然在设计或原型阶段;
- 仅有 10 个系统进行了真实用户实验评估。

另一有趣的发现是,目前并没有针对推荐算法的实验调查,而这种评估在其他领域系统检测中很常见(如[7, 22, 37, 74])。因为在一个推荐系统最终应用到实际场景中之前,必须要进行仔细的检验并将算法参数化(见第 8 章和第 10 章)。这样做的主要原因是,推荐算法表现依赖于特定的应用场景,因而建议在实际应用前,先对多种推荐系统的设计选择进行实验性分析。

## 12.5 TEL 推荐系统的评估

Worthen 等[111]将评估定义为“对标准进行界定、说明、应用来决定评估目标的价值、质量、效用、有效性、或者在这些标准上的重要性”。对交互系统的评估可以确保其

表现符合设计者期望,且能满足用户需求[23]。对推荐系统整体而言,尤其是 TEL 推荐系统,评估已经成为系统成长和成功生命周期中的关键节点。尽管在评估中用户相关评估方法的重要性被一再强调(参见[38, 69]和第 8 章),然而时至今日,推荐系统评估仍注重那些来自信息获取研究中的非常“技术性”的测量方法。教育场景存在的特殊性,使得 TEL 推荐系统评估要求非常高。为此,我们尝试在本章采用不同的视角来总结相关的评估需求。

### 12.5.1 对组件的评估

AEH 系统评估常被认为难度较高,主要是由于以下两类问题[108]:

- 首先,很难确定哪些参照变量可以用来评估系统自适性,因为部分系统可能无法关闭自适应能力,或者系统本身具有自适应性,不存在非自适应版本[40]。在 TEL 推荐系统中,如果能够测量推荐关闭后对感知有用性的影响,那么变量就可以得到确认。
- 其次,对自适性并没有一个合适的界定标准,或者说很少有被公认的合适的标准:一方面,客观指标(如持续时间、交互步骤数量、获取的知识)往往不能体现一个系统自适和非自适版本的区别。另一方面,人机交互研究中使用到的主观标准(如可用性问卷)很少用到自适系统的测量上。在 TEL 上,这一问题和定义合适的评估推荐策略成功与否的方法(如技术、度量、工具)有关。

在这些评估中常见的一个问题是将自适过程看作一个整体进行分析[11],无法提供低粒度的结果,因而实用性较低,也不能为设计师提供系统局部(如用户建模方法、领域建模方法、推荐技术)改进的线索。Brusilovsky 等[13]提出了一个很有趣的方法:将自适过程解构为两个层级,分开进行评估。这一方法背后的核心思想是,评估一个自适系统时,不能将自适应看成一个整体的/单独的流程,而是要分解成为不同的组合部分,如果必要和可行,要对每一个部分分开评估[46]。这一观念最早见于 Totterdell 和 Boyle [94],他们提出了一些自适应测量指标,对应的是自适应用户界面逻辑模型的不同组成部分,以此指导自适应导向设计。此外, Totterdell 和 Boyle 还展示了两种评定方法,验证了“用户模型的成功”这一说法(注意:在他们这个案例中,用户模型也有自适应决策功能):

“用户模型构成了两种评估方法:一种是模型对用户困难推断的准确度评估;另一种是对界面变化的有效性评估。”[94]

与两个层级评估自适应的观点[13]同时提出的还有另外两种层级式(或者称为模块式)的评估框架。第一种是 Weibelzahl[106]提出的基于过程的框架,它包含 4 个层级,这四个层级与自适应过程中的信息处理步骤相对应,包括数据输入评估、推理机制评估、自适决策评估和整体交互评估。第二种是 Paramythi 等[75]提出的框架,它对自适应过程中所包含的不同组成部分划分得更细致,同时还讨论了对于不同自适应“模块”的合适的评估方法与工具用以辅助开发。最终这两种框架的融合版本把两种自适应系统评估标准都考虑了进去,同时也提出了相对应的方法和所需的工具[109]。一些自适应系统评估研究拓展了这一模式化的评估方法(如[15]),但还没有正式开发和应用到推荐系统中,因此,应用到 TEL 推荐之前,需要对其效果进行检验。尽管如此,我们仍然认为在评估 TEL 推荐系统时,可以将这种方法整合进丰富的视角与测量中。接下来我们将对相关问题进行详细阐述。

### 12.5.2 评估 TEL 推荐系统时需要考虑的问题

在所有的用户推荐系统中,一小部分具有特殊属性的数据集是开放的(如 MovieLens 数据集、Book-Crossing 数据集或者 Jester 数据集),这些数据集常用作评估新推荐算法的标准。衡量推荐系统算法的一些常见的技术性测量方法有准确度、覆盖率、执行时间表现等。

而在 TEL 应用领域,对于正式和非正式学习的教育驱动推荐系统评估,目前还没有可用的标准化的数据集和评估程序。只关注 TEL 推荐系统的技术测量,而不考虑学习者实际的需求与特征,这种做法很值得商榷。因此需要更多的评估程序对现有的技术测量方法进行补充(参见第 8 章)。举例来说,只有在系统让学习过程变得更有效、高效,或者更有吸引力时,学习者才能从这种由 TEL 支持与增强的系统中获益。在教育情景下,衡量系统成功的常见指标有有效性、效率、满意度和退出率。有效性标记的是一个学习时段内内容完成、访问或者学习的总体数量。效率指的是学习者完成其学习目标所需的时间,与计算实际学习时间中的有效度变量相关。满意度反映的是学习者对给定推荐的满意程度,满意度与学习者的学习动力紧密相关,因而是一个重要测量指标。最后,退出率反映了在学习期间退出学习的人数。这一指标在教育情境下很重要,因为其目标是要让尽可能多的人毕业。就学习网络而言,来源于社会网络分析(SNA)(如[105])中的方法和指标也可以用来衡量 TEL 推荐系统的成功。SNA 指标可用于估算学习者的贡献给网络整体带来的益处。还有一些特定的指标,主要针对的是非正式学习网络。SNA 能帮助发现学习者在学习网络中所起的多样性作用。典型的 SNA 指标有多样性、中心度、紧密度和内聚度。多样性通过组合个体学习路径到最成功的学习路径来测量学习网络的涌现水平。中心度是某一个学习者在学习网络中的连通性,即该学习者与网络中其他学习者之间所形成的连接数量。紧密度指的是某一学习者与网络中其他学习者的紧密程度,它代表了通过与其他网络成员连接来直接/间接获取信息的能力。内聚度,从另一方面来说,指的是学习者彼此之间在多大程度上存在强关联的程度。如果学习网络中每一个学习者都与其他所有学习者直接相连,则可以认定为同伴学习者群体。Drachsler 等[24]在模拟环境中使用这一方法来评估非正式学习网络中推荐系统对学习者的影响(参见第 18 章)。

将所有这些不同成分整合到一个总括性的评估框架还存在一些方法论和实际操作上的困难。一个整体的方针是,可以采纳教育研究中经典评估框架,并适配到推荐系统场景中。我们将介绍如何将使用 4 种不同层级测量的 Kirpatrick 模型[50]应用到对 TEL 情境推荐系统测量中:

- **用户反应**——他们有何想法和感受(“我喜欢收到的推荐吗?”);
- **学习**——获得的知识或者能力上的提升(“在推荐系统帮助下,我学到了我需要的东西,获得了一些新的想法吗?”);
- **行为**——获取的知识和能力可应用到现实生活中的程度(“我会实际利用这些推荐的新信息和新概念吗?”);
- **结果**——对用户在学习/工作中表现的影响(“推荐给我的信息和概念能帮助提高效能和结果吗?”)。

因此,对 TEL 推荐系统整体评估框架的定义应包含:

- 对评估方法和工具的具体分析,这些方法和工具依据一系列标准来对 TEL 推荐系



统技术进行评价,未来也可以用于对某些组成部分(如用户模型、领域模型、推荐策略和算法)的评估中。对于上面给出的 Kirckpatric 维度的案例,应该包含在四个维度上,对测量 TEL 情境中推荐效果评估方法的明确定义。

- 测量每一部分是否成功(如测量推荐算法的准确性、领域模型的覆盖率)的评估标准/指标的规格。该案例中,应该包含测量 TEL 引入推荐系统效果的特定指标规格。
- 对于 TEL 中的方法和工具的详细阐述,目的在于获取参与者显性或隐性的评估数据,如测量用户满意度、分析使用 TEL 推荐系统对工作提升的影响等。在这一案例中,应该列出用于测量 TEL 推荐系统效果每一指标的工具。

## 12.6 总结与展望

本章介绍在 TEL 环境部署推荐系统的相关问题,并强调了应用领域的特性。首先讨论部署 TEL 推荐系统的上下文,并考虑了相关的用户任务及推荐目标。相关工作的综述来自之前介绍过的自适应教育超媒体和学习网络,特别强调它如何使用在 TEL 推荐的正式和非正式环境中。然后,文献中针对 TEL 推荐的调研提出了一个有关系统实际实现的重要观点。其中特别强调了评估,这引发了评估 TEL 推荐系统需求和问题的讨论,据我们所知,这是首次有研究试图系统地覆盖 TEL 环境的推荐系统的设计(参考第 11 章)。然而,它仅能提供一个相关问题的简要概述,留下几个方面还需要有更多的探索和研究。

本章前一部分的结果表明,主要的研究挑战之一是在 TEL 中引入推荐系统该如何完成系统开发,并评价它们的效果。因此产生的是能够支持推荐系统相关 TEL 任务的系统分析,这样就可以明确需要被考虑的特殊需求。此外,如何开发遵循分层方法的具体评价框架,是一个开放的问题。这些框架通过结合多种评价方法、度量和手段,可以专注于尽可能纳入很多的评价维度,也包括教学上的维度。

此外,对于 TEL 领域的各个研究小组来说,一些课题具有很高的研究价值。例如,推荐学习者在正式和非正式的学习中充分利用情境化推荐系统的支持已经成为一个重要的方式。这些推荐系统也叫作情景感知推荐系统(参考[61]和第 7 章),例如,使用地理位置为用户推荐相关的资源,这种语境化变得重要;从不同国家的大量资料库中推荐多种语言的教育资源,并遵守不同课程的要求[101]。此外,教学情境感知应该包括教学法方面的内容,如先验知识、学习目标或学习时间,并将这些教学推理嵌入协同过滤驱动的推荐结果中。

另一种有前景的方法是在 TEL 推荐系统中使用多标准输入(参考第 24 章)。用户(学生和老师)不仅可以基于复杂程度、课程定位或完成学习材料所需的时间,而且也可以从不同的隐式资源推断出来要选择学习的资源。这种多维的输入对推荐的适用性可能有很高的冲击。在正式和非正式学习中,相关的问题是 TEL 缺乏特定的数据集。在不同的推荐系统中,有很多数据集能被使用(比如,MovieLens、BookCrossing、Jester Collaborative Filtering Dataset),TEL 科研群体仍然工作在较小的自制的数据集上,很少对外公开[66]。

## 致谢

N. Manouselis 的研究由欧洲委员会资金支持,更具体的是 ECP-2006-EDU-410012 项目“组织-教育指导网络:包含关于有机农业和农业生态学的欧洲青年认识和教育的高质内容学习库的多语言联盟”的 eContentplus 计划。H. Drachsler 的研究也是由欧洲委员会资金支持的,更具体的是 IST 027087 ‘TENCompetence’ of the FP6 的项目计划。Riina-Vuorikari 感谢 HS-saatio 的奖学金。

## 参考文献

1. Aehnelt M., Ebert M., Beham G., Lindstaedt S., Paschen A.: A Socio-technical Approach towards Supporting Intra-organizational Collaboration. In: Dillenbourg P. and Sprechth M., (eds.) *Times of convergence: Technologies across learning contexts. Proceedings of the 3<sup>rd</sup> European Conference on Technology Enhanced Learning (EC-TEL 2008)*, Maastricht, The Netherlands, LNCS 5192, pp. 33-38, Berlin: Springer (2008).
2. Anderson M., Ball M., Boley H., Greene S., Howse N., Lemire D., McGrath S.: RACOFI: A Rule-Applying Collaborative Filtering System. Paper presented at the conference IEEE/WIC COLA 2003, 13 October, Halifax, Canada.
3. Aroyo, L., Mizoguchi, R., and Tzolov, C.: OntoAIMS- Ontological Approach to Courseware Authoring. Paper presented at the International Conference on Computers in Education (ICCE 2003), 2-5 December, Hong Kong, China
4. Avancini H., Straccia U.: User recommendation for collaborative and personalised digital archives, *International Journal of Web Based Communities* 1(2), 163-175 (2005).
5. Baldoni, M., Baroglio, C., Brunkhorst, I., Marengo, E., Patti, V.: Reasoning-Based Curriculum Sequencing and Validation: Integration in a Service-Oriented Architecture. In *Proceedings of the 2<sup>nd</sup> European Conference on Technology Enhanced Learning (EC-TEL 2007)*, LNCS 4753, pp. 426. Berlin: Springer (2007).
6. Baudisch P.: Dynamic Information Filtering. PhD Thesis, GMD Forschungszentrum Informationstechnik GmbH, Sankt Augustin (2001).
7. Breese, J. S., Heckerman, D., and Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Cooper G. F. and Moral S., (eds.) *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pp. 43-52, Morgan-Kaufmann, Californian, San Francisco (1998).
8. Brockett, R. G., & Hiemstra, R.: *Self-direction in adult learning: perspectives on theory, research, and practice*. London: Routledge (1991).
9. Brusilovsky, P.: Methods and techniques of adaptive hypermedia, *User Modeling and User-Adapted Interaction* 6(2-3), 87-129 (1996).
10. Brusilovsky, P.: Adaptive hypermedia. *User Modeling and User-Adapted Interaction* 11(1-2), 87-110 (2001).
11. Brusilovsky, P. and Eklund, J.: A study of user-model based link annotation in educational hypermedia, *Journal of Universal Computer Science* 4(4), 429-448 (1998).
12. Brusilovsky P., Nejd W.: *Adaptive Hypermedia and Adaptive Web. Practical Handbook of Internet Computing*, Chapman & Hall / CRC Press LLC (2005).
13. Brusilovsky, P., Karagiannidis, C., and Sampson, D. G.: The benefits of layered evaluation of adaptive applications and services. In: Weibelzahl, S., Chin, D. N., and Weber, G. (eds.) *Empirical Evaluation of Adaptive Systems. Proceedings of 8th International Conference on User Modeling, (UM2001)*, pp 1-8, Berlin: Springer (2001).
14. Brusilovsky, P., Pesin, L., & Zyryanov, M.: Towards an adaptive hypermedia component for an intelligent learning environment. In: Bass, L.J., Gornostae, J., & Unger, C. (eds.) *Human-Computer Interaction (LNCS 753)*. pp. 348-358, Berlin: Springer (1993).
15. Brusilovsky P., Karagiannidis C., Sampson D., Layered evaluation of adaptive learning systems. *International Journal of Continuing Engineering Education and Lifelong Learning* 14(4/5), 402-421 (2004).
16. Brusilovsky, P., & Henze, N.: Open Corpus Adaptive Educational Hypermedia. In P. Brusilovsky, A. Kobsa & W. Nejd (eds.) *The Adaptive Web: Methods and Strategies of Web Personalization*. (LNCS 4321), pp. 671-696 Berlin: Springer (2007).
17. Colley, H., Hodgkinson, P., & Malcolm, J.: Non-formal learning: mapping the conceptual terrain. A consultation report. (2008). Accessed 11 January 2010.
18. Cristea, A.: Authoring of Adaptive Hypermedia. *Educational Technology & Society* 8(3), 6-8 (2005).
19. De Bra, P. M. E.: Teaching Hypertext and Hypermedia through the Web. *Journal of Universal Computer Science* 2(12), 797-804 (1996).
20. De Bra, P., Aerts, A., Smits, D., & Stash, N.: AHA! Version 2.0, More Adaptation Flexibility for Authors. Paper presented at the World Conference on e-Learning in Corporate, Government, Healthcare & Higher Education. 15-19 October 2002, Montreal, Canada (2002).
21. De La Passardiere, B., & Dufresne, A.: Adaptive navigational tools for educational hypermedia. In: I. Tomek (ed.), *Computer Assisted Learning. Proceedings of the 4th Interna-*

- tional Conference on Computers and Learning (ICCAL'92), LNCS 602, pp. 555-567, Berlin: Springer (1992).
22. Deshpande, M., & Karypis, G.: Selective Markov models for predicting Web page accesses. *Transactions on Internet Technology*. 4(2), 163-184 (2004).
23. Dix, A. J., Finlay, J. E., Abowd, G. D., and Beale, R.: *Human-Computer Interaction*. Harlow, England: Prentice Hall (1998).
24. Drachsler H., Hummel H. G. K., Koper R.: Personal recommender systems for learners in lifelong learning: requirements, techniques and model. *International Journal of Learning Technology* 3(4), 404-423 (2008a).
25. Drachsler, H., Hummel, H.G.K., Koper, R.: Using Simulations to Evaluate the Effects of Recommender Systems for Learners in Informal Learning Networks. In: Vuorikari, R., Kieslinger, B., Klamma, R., Duval, E. (eds.): *SIRTEL workshop at the 3rd EC-TEL conference*. *CEUR Workshop Proceedings VOL-382*, Maastricht, The Netherlands (2008b).
26. Drachsler, H., Pecceu, D., Arts, T., Hutten, E., Rutledge, L., Van Rosmalen, P., Hummel, H.G.K., Koper, R.: ReMashed-Recommendations for Mash-Up Personal Learning Environments. In: Cress, U., Dimitrova, V., Specht, M. (eds.): *Learning in the Synergy of Multiple Disciplines, Proceedings of the 4<sup>th</sup> European Conference on Technology Enhanced Learning (EC-TEL 2009)*, LNCS 5794, pp. 788-793, Berlin: Springer (2009a).
27. Drachsler, H., Pecceu, D., Arts, T., Hutten, E., Rutledge, L., Van Rosmalen, P., Hummel, H.G.K., Koper, R.: ReMashed-An Usability Study of a Recommender System for Mash-Ups for Learning. 1st Workshop on Mashups for Learning at the International Conference on Interactive Computer Aided Learning, Villach, Austria (2009b).
28. Drachsler, H., Hummel, H. G. K., Van den Berg, B., Eshuis, J., Berlanga, A., Nadolski, R., Waterink, W., Boers, N., & Koper, R.: Effects of the ISIS Recommender System for navigation support in self-organized learning networks. *Journal of Educational Technology and Society* 12, 122-135 (2009c).
29. Dron, J., Mitchell, R., Boyne, C., & Siviter, P.: CoFIND: steps towards a self-organising learning environment. *Proceedings of the World Conference on the WWW and Internet (WebNet 2000)*, San Antonio, Texas, USA, October 30-November 4, pp. 146-151, USA AACE (2000a).
30. Dron, J., Mitchell, R., Siviter, P., & Boyne, C.: CoFIND-an experiment in n-dimensional collaborative filtering. *Journal of Network and Computer Applications* 23(2), 131-142 (2000b).
31. Duval E., Vuorikari R., Manouselis N. (Eds.), *Special Issue on Social Information Retrieval in Technology Enhanced Learning*, *Journal of Digital Information (JoDI)*, Vol. 10, (2009).
32. Fiaidhi J.: RecoSearch: A Model for Collaboratively Filtering Java Learning Objects. *International Journal of Instructional Technology and Distance Learning* 1(7), 35-50 (2004).
33. Gomez-Albarran M., Jimenez-Diaz G.: Recommendation and Students' Authoring in Repositories of Learning Objects: A Case-Based Reasoning Approach. *International Journal of Emerging Technologies in Learning (iJET)* 4(1), 35-40 (2009).
34. Gonschorek, M., & Herzog, C.: Using hypertext for an adaptive helpsystem in an intelligent tutoring system. In: J. Greer (Ed.), *Artificial Intelligence in Education, Proceedings of the 7th World Conference on Artificial Intelligence in Education, (AI-ED'95)*, 16-19 August, Washington, DC, AACE, pp. 274 (1995).
35. Gordon, D.: *Ants at Work: How an Insect Society is Organized*. Free Press, New York (1999).
36. Graham, A.: Blended learning systems: definitions, current trends and future directions. In: Bonk, C. J. & Graham, C. R. (eds.) *Handbook of blended learning: Global Perspectives, local designs*. San Francisco, CA: Pfeiffer Publishing, pp. 3-21 (2005).
37. Herlocker, J., Konstan, J., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information Retrieval* 5(4), 287-310 (2002).
38. Herlocker, J.L., Konstan, J.A., Terveen, L.G., & Riedl, J.T.: Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems* 22(1), 5-53 (2004).
39. Hohl, H., Böcker, H.-D., Gunzenhäuser, R.: Hypadapter: An adaptive hypertext system for exploratory learning and programming. *User Modeling and User-Adapted Interaction* 6(2-3), 131-156 (1996).
40. Höök, K.: Steps to take before intelligent user interfaces become real. *Interacting With Computers* 12(4), 409-426 (2000).
41. Huang, Y.-M., Huang, T.-C., Wang, K.-T., Hwang, W.-Y.: A Markov-based Recommendation Model for Exploring the Transfer of Learning on the Web. *Educational Technology & Society* 12(2), 144-162 (2009).

42. Hummel, H.G.K., Van den Berg, B., Berlanga, A.J., Drachsler, H., Janssen, J., Nadolski, R.J., Koper, E.J.R.: Combining Social- and Information-based Approaches for Personalised Recommendation on Sequencing Learning Activities. *International Journal of Learning Technology* 3(2), 152–168 (2007).
43. Jameson, A.: *Systems That Adapt to Their Users: An Integrative Perspective*. Saar-brücken: Saarland University (2001).
44. Janssen, J., Tattersall, C., Waterink, W., Van den Berg, B., Van Es, R., Bolman, C., et al.: Self-organising navigational support in lifelong learning: how predecessors can lead the way. *Computers & Education* 49, 781–793 (2005).
45. Johnson, S.: *Emergence*. Scribner, New York (2001).
46. Karagiannidis, C., Sampson, D. G.: Layered evaluation of adaptive applications and services. In: Brusilovsky, P. and Stock, C. S. O. (Eds.), *Proc. of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, AH2000, Trento, Italy*, pp. 343–346. Berlin: Springer (2000).
47. Karampiperis, P., Sampson, D.: Adaptive Learning Resources Sequencing in Educational Hypermedia Systems. *Educational Technology & Society* 8(4), 128–147 (2005).
48. Kay, J., Kummerfeld, R. J.: An individualised course for the C programming language. In: *Proceedings of Second International WWW Conference*. Chicago, USA (1994).
49. Khribi, M.K., Jemni, M., Nasraoui, O.: Automatic Recommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval. *Educational Technology & Society* 12(4), 30–42 (2009).
50. Kirkpatrick, D.L.: *Evaluating Training Programs* (2nd ed.). Berrett Koehler, San Francisco (1959).
51. Klamma, R., Spaniol, M., Cao, Y.: Community Aware Content Adaptation for Mobile Technology Enhanced Learning. In: *Innovative Approaches for Learning and Knowledge Sharing*, pp. 227–241 (2006).
52. Koper, R.: Increasing Learner Retention in a Simulated learning network using Indirect Social Interaction. *Journal of Artificial Societies and Social Simulation*, 8(2) (2005).
53. Koper, E.J.R., Tattersall, C.: New directions for lifelong learning using network technologies. *British Journal of Educational Technology* 35(6), 689–700 (2004).
54. Koper, R., Rusman, E., & Sloep, P.: Effective Learning Networks. *Lifelong Learning in Europe* 1, 18–27 (2005).
55. Koutrika, G., Ikeda, R., Bercovitz, B., Garcia-Molina, H.: Flexible Recommendations over Rich Data. In: *Proc. of the 2nd ACM International Conference on Recommender Systems (Rec-Sys'08)*. Lausanne, Switzerland, (2008).
56. Koutrika, G., Bercovitz, B., Kaliszan, F., Liou, H., Garcia-Molina, H.: CourseRank: A Closed-Community Social System Through the Magnifying Glass. In: *Proc. of the 3rd International AAAI Conference on Weblogs and Social Media (ICWSM'09)*. San Jose, California (2009).
57. Kravcik, M., Specht, M., Oppermann, R.: Evaluation of WINDS Authoring Environment. In: P. De Bra & W. Nejdl (Eds.), *Adaptive Hypermedia and Adaptive Web-Based Systems, LNCS 3137*, 166–175. Berlin: Springer (2004).
58. Krulwich, B.: Lifestyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data. *Artificial Intelligence Magazine* 18(2), 37–45 (1997).
59. Kumar, V., Nesbit, J., Han, K.: Rating Learning Object Quality with Distributed Bayesian Belief Networks: The Why and the How. In: *Proc. of the Fifth IEEE International Conference on Advanced Learning Technologies, ICALT'05* (2005).
60. Lemire, D.: Scale and Translation Invariant Collaborative Filtering Systems. *Journal of Information Retrieval* 8(1), 129–150 (2005).
61. Lemire, D., Boley, H., McGrath, S., Ball, M. (2005). Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation. *International Journal of Interactive Technology and Smart Education* 2(3) (2005).
62. Liber, O.: Colloquia - a conversation manager. *Campus-Wide Information Systems* 17, 56–61 (2000).
63. Liber, O., Johnson, M. (2008). Personal Learning Environments. *Interactive Learning Environments* 16, 1–2 (2008).
64. Longworth, N.: *Lifelong learning in action - Transforming education in the 21st century*. Kogan Page, London (2003).
65. Manouselis, N., Costopoulou, C.: Experimental Analysis of Design Choices in Multi-Attribute Utility Collaborative Filtering. *International Journal of Pattern Recognition and Artificial Intelligence, Special Issue on Personalization Techniques for Recommender Sys-*

- tems and Intelligent User Interfaces 21(2), 311–331 (2007).
66. Manouselis, N., Vuorikari, R.: What if annotations were reusable: a preliminary discussion. In: Proc. of the 8th International Conference on Web-based Learning (ICWL 2009). Aachen, Germany (2009).
67. Manouselis, N., Vuorikari, R., Van Assche, F.: Simulated Analysis of MAUT Collaborative Filtering for Learning Object Recommendation. In: Proc. of the Workshop on Social Information Retrieval in Technology Enhanced Learning (SIRTEL 2007). Crete, Greece (2007).
68. McCalla, G.: The Ecological Approach to the Design of E-Learning Environments: Purpose-based Capture and Use of Information About Learners. *Journal of Interactive Media in Education*, Special Issue on the Educational Semantic Web, 7, ISSN:1365-893X (2004).
69. McNee, S.: Meeting User Information Needs in Recommender Systems. Doctoral dissertation, University of Minnesota-Twin Cities, Minneapolis, MN, USA (2006).
70. Moore, J.D., Swartout, W.R.: Pointing: A way toward explanation dialogue. In: Proceedings of the Eighth National Conference on Artificial Intelligence, pp. 457–464. AAAI (1990).
71. Moore, M.G., Anderson, W.G.: Handbook of distance education. Lawrence Erlbaum, Mahwah N.J. (2004).
72. Nadolski, R.J., Van den Berg, B., Berlanga, A., Drachsler, H., Hummel, H., Koper, R., Sloep, P.: Simulating Light-Weight Personalised Recommender Systems in Learning Networks: A Case for Pedagogy-Oriented and Rating-Based Hybrid Recommendation Strategies. *Journal of Artificial Societies and Social Simulation (JASSS)*, 12(14) (2009).
73. Oppermann, R.: Adaptively supported adaptability. *Int. J. Hum.-Comput. Stud.* 40(3), 455–472 (1994).
74. Papagelis, M., Plexousakis, D., Kutsuras, T.: Alleviating the Sparsity Problem of Collaborative Filtering Using Trust Inferences. In: Proceedings of the 3rd International Conference on Trust Management, pp. 224–239. Springer (2005).
75. Paramythis, A., Totter, A., Stephanidis, C.: A modular approach to the evaluation of adaptive user interfaces. In Weibelzahl, S., Chin, D., Weber, G. (eds.) *Empirical Evaluation of Adaptive Systems. Proceedings of workshop at the Eighth International Conference on User Modeling, UM2001*, pp. 9–24. Freiburg (2001).
76. Paris, C.: Tailoring object description to a user's level of expertise. *Computational Linguistics* 14(3), 64–78 (1988).
77. Pérez, T., Gutiérrez, J., Lopistéguy, P.: An adaptive hypermedia system. In: Proceedings of AI-ED'95, 7th World Conference on Artificial Intelligence in Education, pp. 351–358. AACE (1995).
78. Rafaeli, S., Barak, M., Dan-Gur, Y., Toch, E.: QSIA-a Web-based environment for learning, assessing and knowledge sharing in communities. *Computers, Education* 43(3), 273–289 (2004).
79. Rafaeli, S., Dan-Gur, Y., Barak, M.: Social Recommender Systems: Recommendations in Support of E-Learning. *International Journal of Distance Education Technologies* 3(2), 29–45 (2005).
80. Recker, M.M., Walker, A.: Supporting "Word-of-Mouth" Social Networks through Collaborative Information Filtering. *Journal of Interactive Learning Research* 14(1), 79–99 (2003).
81. Recker, M.M., Wiley, D.A.: A non-authoritative educational metadata ontology for filtering and recommending learning objects. *Interactive learning environments* 9(3), 255–271 (2001).
82. Recker, M.M., Walker, A., Wiley, D.: An interface for collaborative filtering of educational resources. In: *International Conference on Artificial Intelligence*, pp. 26–29. Las Vegas, Nevada, USA (2000).
83. Recker, M.M., Walker, A., Lawless, K.: What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education. *Instructional Science* 31(4/5), 299–316 (2003).
84. Rokach, L., Maimon, O., Averbuch, M.: Information Retrieval System for Medical Narrative Reports, Lecture Notes in Artificial intelligence 3055, page 217–228 Springer-Verlag (2004)
85. Rokach, L., Maimon, O., Arbel, R.: Selective voting-getting more for less in sensor fusion, *International Journal of Pattern Recognition and Artificial Intelligence* 20 (3), pp. 329–350 (2006)
86. Santos, O.C.: A recommender system to provide adaptive and inclusive standard-based support along the eLearning life cycle. In: Proceedings of the 2008 ACM conference on Recommender systems, pp. 319–322 (2008).
87. Schneider-Hufschmidt, M., Kuhme, T., Malinowski, U.: *Adaptive User Interfaces: Principles and Practice*. Elsevier Science Inc (1993).



88. Shen, L., Shen, R.: Learning content recommendation service based-on simple sequencing specification. In: Liu W et al. (eds), pp. 363-370. Lecture notes in computer science (2004).
89. Tang T., McCalla G.: Smart Recommendation for an Evolving E-Learning System. In: Proc. of the Workshop on Technologies for Electronic Documents for Supporting Learning, International Conference on Artificial Intelligence in Education (AIED 2003) (2003).
90. Tang, T., McCalla, G.: Utilizing Artificial Learner on the Cold-Start Pedagogical-Value based Paper Recommendation. In: Proc. of AH 2004: International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (2004a).
91. Tang, T., McCalla, G.: Beyond Learners' Interest: Personalized Paper Recommendation Based on Their Pedagogical Features for an e-Learning System. In: Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence (PRICAI 2004), pp. 301-310 (2004b).
92. Tang, T.Y., McCalla, G.: On the pedagogically guided paper recommendation for an evolving web-based learning system. In: Proceedings of the 17th International FLAIRS Conference, pp. 86-91 (2004c).
93. Tang, T., McCalla, G.: Smart Recommendation for an Evolving E-Learning System: Architecture and Experiment. *International Journal on E-Learning* 4(1), 105-129 (2005).
94. Totterdell, P., Boyle, E.: The evaluation of adaptive systems. *Adaptive User Interfaces* 161-194 (1990).
95. Tsai, K.H., Chiu T.K., Lee M.C., Wang T.I.: A learning objects recommendation model based on the preference and ontological approaches. In: Proc. of 6th International Conference on Advanced Learning Technologies (ICALT'06). IEEE Computer Society Press (2006).
96. Tzikopoulos, A., Manouselis, N., Vuorikari, R.: An overview of Learning Object Repositories. In: *Learning Objects for Instruction, Design and Evaluation*, pp. 29-55. Idea Group Publishing, Hershey, PA (2007).
97. Van Bruggen, J., Sloep, P., van Rosmalen, P., Brouns, F., Vogten, H., Koper, R., Tattersall, C.: Latent semantic analysis as a tool for learner positioning in learning networks for lifelong learning. *British Journal of Educational Technology* 35(6), 729-738 (2004).
98. Van Setten, M.: Supporting people in finding information: hybrid recommender systems and goal-based structuring. *Telematica Instituut Fundamental Research Series NO. 016 (TI/FRS/016)*. Enschede, The Netherlands (2005).
99. Vuorikari, R., Berendt, B.: Study on contexts in tracking usage and attention metadata in multilingual Technology Enhanced Learning. In: Fischer, S., Maehle, E., Reischuk, R. (eds.) *Im Focus das Leben*, pp. 181, 1654-1663. Lecture Notes in Informatics (2009).
100. Vuorikari, R., Koper, R.: Ecology of social search for learning resources. *Campus-Wide Information Systems* 26(4), 272-286 (2009).
101. Vuorikari, R., Ochoa, X.: Exploratory Analysis of the Main Characteristics of Tags and Tagging of Educational Resources in a Multi-lingual Context. *Journal of Digital Information* 10(2) (2009).
102. Vygotsky, L.: *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press (1978).
103. Waldrop, M.: *Complexity: The Emerging Science at the Edge of Order and Chaos*. Simons, Schuster, New York (1992).
104. Walker, A., Recker, M., Lawless, K., Wiley, D.: Collaborative information filtering: A review and an educational application. *International Journal of Artificial Intelligence in Education* 14(1), 3-28 (2004).
105. Wasserman, S., Faust, K.: *Social network analysis: Methods and applications*. Cambridge Univ Pr (1994).
106. Weibelzahl, S.: Evaluation of adaptive systems. In: *User Modeling: Proceedings of the Eighth International Conference, UM2001*, pp. 292-294 (2001).
107. Weibelzahl, S. Evaluation of adaptive systems. PhD dissertation. University of Trier, Germany (2003).
108. Weibelzahl, S., Paramythi, A., Totter, A.: A layered framework for the evaluation of interactive adaptive systems. In: Proc. 2nd Workshop on Empirical Evaluation of Adaptive Systems. Johnstown (2003).
109. Wild, F., Mödritscher, F., Sigurdarson, S.E.: Designing for change: mash-up personal learning environments. *eLearning Papers*. 9 (2008).
110. Wilson, S., Sharples, P., Griffiths, D.: Distributing education services to personal and institutional systems using Widgets. In: *Mash-Up Personal Learning environments, Proceedings of the 1st MUPPLE workshop*. CEUR-Proceedings (2008).
111. Worthen, B.R., Sanders, J.R., Fitzpatrick, J.L.: *Program evaluation*. Longman, New York (1997).



## 推荐系统的影响

- 第 13 章 基于评价推荐系统的进展
- 第 14 章 构建更值得信任和具有说服力的推荐系统：特性对评估推荐系统的影响
- 第 15 章 设计和评估推荐系统的解释
- 第 16 章 基于实例评价研究的产品推荐系统的可用性准则
- 第 17 章 基于示意图的产品目录可视化

# 基于评价推荐系统的进展

Lorraine McGinty 和 James Reilly

**摘要** 在过去的 10 年,大量的推荐系统研究表明,采用基于评价接口(如“向我展示类似于商品 A 且更便宜的商品”)的会话架构带来诸多好处。随着推荐系统具有越来越强烈的需要去帮助用户摆脱在众多产品中进行选择时的困扰,基于评价的推荐方式引起更多人的兴趣。最初提出的基于评价的方式是为了解决会话推荐中发现偏好这个核心问题,而如今,评价方式已经在如群推荐、混合主动推荐、自适应用户界面、推荐解释等许多相关领域中成为一个热门的话题。本章对近几年评价研究工作进行了一个非技术性概述,旨在为刚进入本领域的研究者和实践者提供一个综合且简洁的参考来源。

## 13.1 简介

评价研究领域的改进很大程度上受到了在线用户的需求变化和待探索的复杂产品领域的影响。过去 10 年,多种基于评价的推荐方法被提出,并且这些方法在提高推荐效果上的潜力也受到了证实。值得商榷的是,一项研究重要性的衡量,不但依据其能解决现存的问题的数量(例如,提出解决已知问题的方法),而且也依据其提出问题的数量,这些问题的答案作为下一步研究的目标。

本章围绕早期研究于这个领域中的研究者,旨在为近年研究的焦点之一——评价方式及其所能达成的效果与存在的陷阱——提供一个有用的且深入的非技术性概述。在 13.2 节,我们描述了一些在早期评价系统中的做法,以及采用此类方法所获得的好处。同时,在 13.3 节列出了一些需要改进的关键问题和挑战:1)基于优化偏好获取的评价展现;2)基于评价的推荐系统的检索性能。13.4 节对影响跨平台反馈和用户环境下集成评价模型时的各种设计考虑因素进行概述,其中用户环境包括:移动平台、个人和协作平台等。13.5 节概括了一些被业界人士采纳并沿用至今的资源、方法和评价标准。13.6 节概括了一些现存的基于评价推荐的挑战和机会。

## 13.2 早期:评价系统/已得益处

交互推荐系统通常让用户参与到一个交互的对话中,以此来得知他们的偏好,并使用用户对话中的反馈信息来提高系统的推荐精准度。许多会话推荐系统利用实际中使用的例子来发现偏好(详情参见第 21 章)。比如,给一个用户展示一个或者多个案例/推荐品(例如,一部电影、书、照相机),然后让他对展示的物品提供一个反馈(进行评分,或表示是否喜欢)。在进行推荐系统研究和开发基于实例的交互架构中,评价反馈模型已经成为了一个热门的话题。其主要原

---

Lorraine McGinty, UCD School of Computer Science and Informatics, University College Dublin, Dublin 4, Ireland. e-mail: lorraine.mcginity@ucd.ie

James Reilly, Google Inc., 5 Cambridge Center, Cambridge, MA 02142, United States. e-mail: jamesreilly@google.com

翻译:赵华飞,卢青峰,郑州大学-吴宾 审核:李雪,吉林大学-丁彬钊,胡聪(胡户主)

因是该模型在用户提供反馈所需要付出的额外代价和这些反馈信息的价值之间做了一个可接受的折中。与其他标准值抽取方法相比,它只需要用户对当前的需求(比如,“给我展示更多像物品 A 但某特征不同的物品”)提供很少额外的操作来表示一个相对明确的反馈。评价也非常适合最基本的接口界面以及只对推荐领域有基本理解的用户[7, 8, 65]。

在很多情况下,我们不能假设用户起初就完全清楚自己的偏好[48],或者对特征的权衡有一个清晰的认识(相关讨论请见第 11 章)。比如,一位用户想买一台笔记本电脑,开始并没有考虑通常需要在产品的质量和内置 CD-ROM 之间做一个权衡,这时用户如果使用“轻于 2 公斤”而且“带 CD-ROM”的条件去查找,那么可能根本找不到同时满足这两个条件的计算机。而对这个领域的产品特性和可选条件越来越清楚时,用户的偏好通常又发生了改变,变得更加挑剔了[48]。在发现有效信息前,用户往往不会明确表达他们的偏好[67]。事实上,这是不可取的,因为有些时候只需要部分偏好信息就能够为用户提供一个很好的推荐[17]。

基于评价的交互式推荐不但在用户浏览产品类别时提供了灵活的支持,而且能帮助用户更好地了解自己的需求;它不会在一开始就让用户表明他们的偏好,而是在一系列推荐周期中建立用户偏好信息。在每一次推荐会话中,系统为用户推荐一个或多个物品,并邀请他们对其中一个物品进行评价。特征的评价有两种典型的形式:有向或替代(在[37]已定义的名词)。有向评价会影响一个量化特征价值的增加或降低,比如,“便宜”表示少于某价格。替代评价允许用任何值(除了被评价的值)来替换非量化特征,比如,“不同的制造商”指代“!= 制造商”。返回满足所选评价标准的推荐项,并邀请用户按其需求对此推荐进行评价。这种处理方式会持续到以下几种情况的发生:1)用户接受推荐;2)所有潜在的可能结果都已经展示过了;3)用户过早地结束了这个会话。

早期的工作要追溯到 20 世纪 80 年代初期,RABBIT 系统[70, 71]将评价机制作为构造数据库查询的一个新的接口范例进行了介绍。用户能够使用类似禁止或者设限等选择项来评价实例记录的域。在用户反馈的基础上,系统可以重新构造查询条件,然后展示另外的样例记录。由 Burke 等开发的 FindMe 系统[7, 8, 19],是首个在基于 Web 推荐中使用评价机制,其推荐注重于引导用户去了解可选择的空间<sup>⊖</sup>。起初,FindMe 推荐系统是作为浏览助手被开发的,使用为展示的实例提供评价的方式帮助用户在大量的信息中进行浏览。例如,针对餐馆的 Entrée、针对汽车的 Car Navigator、针对公寓的 RentMe、针对电影租借的 Video Navigator<sup>⊖</sup>等基于实例评价的系统。其他早期的基于实例的评价系统包括:Apt Decision[64]、SmartClient[53, 54, 55]、Automated Travel Assistant(ATA)[22]。在 13.3 节中,我们概述了这些基于评价的系统和后续研究中要解决的关键性挑战。

### 13.3 评价系统的表述与检索挑战

对于早期的系统和近期的评价机制审视方法有很多维度,本节重点关注其中的两个维度:评价的表述与推荐的检索。在第一个例子中,我们对已经用于优化偏好获取的可选评价的展示方式或者构造方法进行了区分,描述了作为反馈选项展示给用户的那些评价属性,是如何表现出迥异的局限和好处。第二个例子中,我们概述了已经在评价机制系统提出来的检索挑战关键性的问题,并讨论了这些问题如何影响更复杂的偏好建模与修改的方法。

#### 13.3.1 评价表述的方式

在过去的 10 年里,人们提出了许多评价展示的替换方案。Entrée[8]餐馆推荐系统是

⊖ 近期的研究强调,用户探求选择空间的需求和期望较短推荐对话之间存在矛盾。

⊖ The Wasabi Personal Shopper[4]是 FindMe 系统的一个通用的、领域无关的版本。



FindMe 系统系列中, 最早被人熟知的成员之一。Entrée 提供了两种初始化用户偏好的方法。一种是指明一个与用户想要找的餐馆相似的已知餐馆, 或者让用户指明他们用餐的兴趣的高层特征来启动推荐会话。高层特征包括 7 个预定义的单一评价选择, 比如, 便宜用于评价价格特征; 更正式用于评价风格特征。评价用于生成下一个推荐周期临时的产品过滤器, 滤除了不在考虑范围内的餐馆。本章讨论一些挑战, 它们促使该领域进一步研究和发展。

### 13.3.1.1 过分的评价与冗长的推荐交互

每个推荐周期中, 单一评价只允许用户表达一个特征上的偏好, 这最终限制了推荐系统的推荐范围, 并导致了很多不必要的推荐交互。此外, 在一些特殊领域, 用户可能不清楚特征之间的权衡关系, 因此他们可能更倾向于继续评价一个特定的特征(如价格), 直到发现一个可以接受的推荐结果。然而, 他们可能会发现另外一个很重要的特征的值被改变后, 推荐结果就不符合预期了[39]。一个替代的策略是考虑使用复合评价(在[26]中首次被介绍), 这些复合评价控制多个特征, 因为在一个推荐周期中收集了多个偏好信息, 所以很有可能提高推荐系统的效率。可以保证的是, 使用复合评价可以使用户离想要的推荐更近, 因此总之减少了会话的长度/交互时间。

其实, 复合评价并不是一个新的想法, 早期的 FindMe 系统[7]中的 Car Navigator 系统(示例见图 13.1)就使用了复合评价的方式。汽车由马力、价格、运动、油耗等特征来描述, 用户能直接对这些特征进行操作。复合评价与单一的特征级别的评价并列展示, 并为用户提供了 2 个可选的途径来改善推荐。举例说明, 当选择运动(sportier)这个复合特征时, 将依据一些特征对预留的选项进行过滤——发动机动力大小、加速和价格这些特征的值都增加了。类似地, 在 PC(个人电脑)的推荐场景中, 复合评价——高性能可能会同时增加处理器速度、内存、硬盘容量和价格等特征的值。



图 13.1 演示 Car Navigator 系统是如何展示固定的单一与组合评价的

### 13.3.1.2 评价冗余与隐藏的特征依赖

由于观察到固定的评价方式不是总能得到相关确切的结果, 所以自动生成评价的概念被提

出来,最先由 McCarthy 等提出[26]。他们认识到,会话死角会导致一个用户使用一个评价(个体的或复合的)进行过滤时发现没有符合条件的推荐结果[18, 37, 39]。比如,在汽车检索系统中,即使没有推荐结果对应评价标准“运动型”的情况下,其仍然作为一个选项被展示出来。这种固定评价方式(包括单一和复合)让用户对这种情况下的可供选择的推荐选项很不理解。事实上,他们完全可以避免这种对用户困扰的情况并去掉多余的交互接口。例如,对这些固定的标签的真实意义,用户的理解不需要完全相同(比如,用户不需要知道采用了什么样的隐藏特征来进行过滤)。不可避免的是,这意味着用户会发现这些固定的复合评价的用处变小了。

McCarthy 等[26]讨论了是否需要一个更加灵活的用于生成评价的动态方法,在每个推荐周期中,这个方法动态地对复合评价进行组合,以此展示可用评价包括:1)更好的重点推荐;2)限制交互冗余;3)消除用户的疑虑。他们证实了一个观点:只有包含可用候选集的复合评价才需要被展示给用户。当然,使用同样的基本理论生成一个个体评价也是合理的,就如 Tweak 系统[39]中实现的那样,确保了只为用户展示个体评价选项,并且有至少一个产品对应于此选项。三种动态生成复合评价的方法被提出[26, 73, 12]。所有的这些方法为用户提供了多特征评价的选择,基于这种评价(例如,推荐结果)的一个应用将会为用户进行特征的改变。例如,一个用户想要看一些与推荐的计算机相似,但是更加便宜、有更大的内存,并且不是同一个生产商生产的计算机,他可以从不同的厂商中选择——更大内存,更少价格。所有提到的这些方法,都被证明在一定程度上对推荐效率、准确率、有效性、可用性等性能产生了有利影响。区别这些方法的关键因素是影响产生评价的知识(比如,领域知识和用户偏好知识的比较)。下面将描述各种方法的不同之处与被提出的原因。

### 13.3.1.3 有限的产品展示空间

Reilly 等[59]在他们的 QwikShop 系统(见图 13.2)中演示了在推荐开始之前,如何只用推荐候选集的领域知识来自动生成复合评价。他们专注于通过加深用户对当前推荐结果的特征协调与相互依赖的认识,来提高他们对用于进行推荐候选集调整的那些特性的理解深度。称为动态复合评价算法,使用了数据挖掘中关联规则的 Aprior 算法[1]来发现推荐待选集中频繁出现的特征关系。通常,比如,“更大分辨率”“更大内存”“不同制造商”(与当前的推荐结果比较)这种形式的大量复合评价是由这种方法论产生的。从用户的角度来看,展示大量的复合评价选项是不可取的。为了在单个的标准评价选项旁,给用户展示指定个数的组合评价,可以采用一些策略来对组合评价选项进行排序,如支持度、置信度、平衡度、提升度和说服力<sup>①</sup>。McCarthy 等[26]与 Reilly 等[56]调查研究了根据支持度的值来进行复合评价的最佳排序(其中的支持度指满足评价条件的产品比例)。他们表明低支持度的组合评价选项在可用性和专注搜索能力之间能找到一个很好的平衡点。

最终选择的组合评价选项,可能会表现出有限的特征多样性(例如,他们的组成个体特征能够高度重叠),因为现有的很多选项对于当前的推荐结果来说,其特征差异很小。推荐系统领域中的人都熟悉的一个问题,即用户选择的推荐集与他们的偏好有关,但又各不相同。与此对应的,相关文献也提出了一些强化多样性的解决方案(见 13.3.2 节),并研究出了一些动态展示多样性的复合评价的方法[28],这些方法生成的反馈选项,使得用户适用性最高提升了 32%。这类方法的潜在限制是,它并没有将用户的偏好考虑进去。在 QuikShop 系统中,用户偏好只影响了推荐结果的选择,并没有作为反馈选项展示给用户。然而,作为用户偏好的关系松散的模型,有更大的潜力来发现有意外效果的推荐结果。

① 这些度量标准是关联规则中常用的[1]。

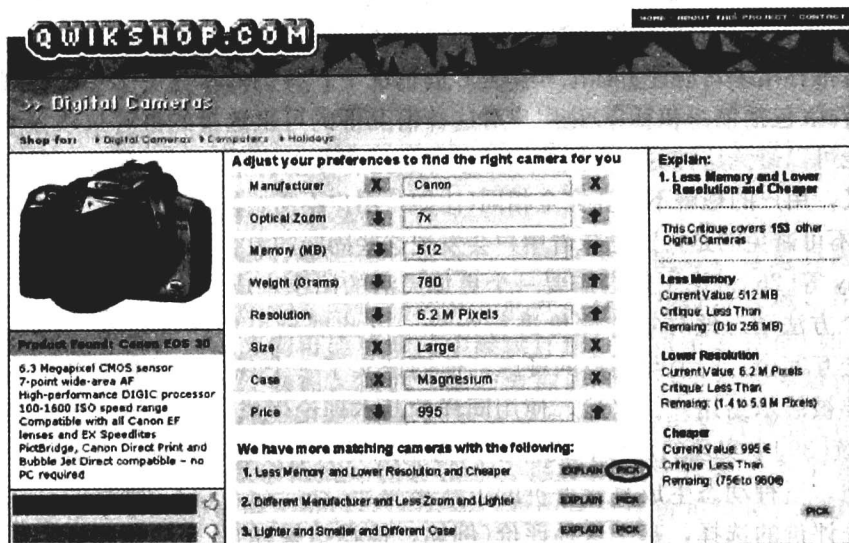


图 13.2 QwikShop 评价系统截图

#### 13.3.1.4 被展示反馈选项之间的弱相关性

如上所述, 根据 Reilly 等的方法所生成的复合评价并不保证与用户的相关性, 因为产生的方式只与产品领域知识有关。Zhang 和 Pu[73]提出了一个依赖用户偏好知识的可选方法。他们的多属性效用理论(Mutil-Attribute Utility Theory, MAUT)与 SmartClient 系统[53, 54, 55]所实现的方法相似同样实现了这个功能, 即用于动态生成复合评价。如图 13.3 所示, 他们的 EasyShop 推荐系统主要是维护基于用户的评价反馈的偏好模型, 并计算所有的待推荐选项的产品效用值。

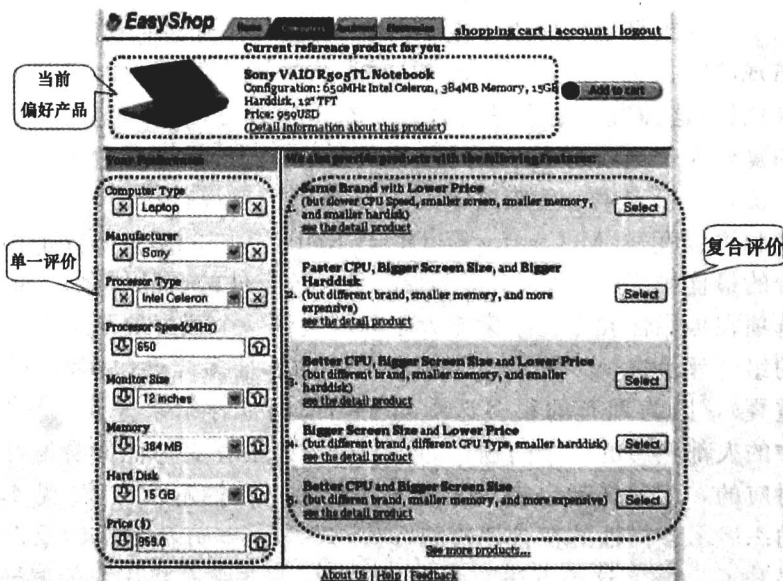


图 13.3 EasyShop 评价系统的截图[73]

具有最高效用值的产品会被选择作为下一个推荐, 并且下一批具有最高效用值的  $k$  个产品的复合评价会作为反馈选项展示出来。离线评估比较现实, 这些面向喜好的复合评价

比 McCarthy 等描述的方法[26]更加接近用户意图评价的标准。另外，他们趋于使用更短但有更高推荐准确度的会话。但是，MAUT 方法也受到一些缺点的困扰。首先，每个由 MAUT 方法生成的复合评价只描述一个产品，因此，这些评价局限了提供给用户更多物品的机会；其次，这种方法假定了一个准确的用户偏好模型。但是，用户的偏好一般是不固定的，而且经常会很快改变(见 13.3.2 节)；最后，它不能在评价中提高多样性，并且在复合评价中更趋于包含大量的特征。

13.3.1.5 领域局限性与偏好驱动的方法

Chen 和 Pu[12]提出了一个动态生成复合评价的方法，目标是保留之前提到的方法的优势，同时最小化其局限性。他们基于偏好的方法目标是动态生成多样的评价选项，这些评价选项适应于用户的偏好，而且代表了待推荐集。

用一个多属性效用模型来表示用户偏好，一个数据挖掘算法生成复合评价集。这些复合评价是最匹配用户的偏好模型的可用待推荐集的分类。在每个复合评价中，提取一些精选的产品来进行展示，使得他们针对当前推荐显示出高权衡效用且相互之间具有差异。一个类别的权衡效用表示了它适应用户模型的程度。直觉上，在当前最好的推荐候选集上，权衡效用越高的类目包含的产品更有潜力使利大于弊(见图 13.4)。

The top candidate according to your preferences									
Manufacturer	Price	MegaPixels	Optical zoom	Memory type	Flash memory	LCD screen size	Depth	Weight	
Canon	\$242.00	5.0 MP	3x	CompactFlash Card	32 MB	1.8 in	1.37 in	8.3 oz	choose
We have some products with the following									
Nikon	\$167.95	4 MP	3x	SD Memory Card	14 MB	1.8 in	1.4 in	4.6 oz	choose
Canon	\$230.00	4.1 MP	3x	CompactFlash Card	32 MB	1.5 in	1.09 in	5.53 oz	choose
Canon	\$180.00	3.3 MP	3x	SD Memory Card	16 MB	2 in	0.83 in	4.06 oz	choose
Canon	\$219.18	4.2 MP	4x	MultiMedia Card	16 MB	1.8 in	1.51 in	5.35 oz	choose
Canon	\$163.50	3.2 MP	4x	MultiMedia Card	16 MB	1.9 in	1.5 in	6.3 oz	choose
Canon	\$199.40	3.2 MP	2.2x	SD Memory Card	16 MB	1.5 in	1.4 in	5.8 oz	choose
Sony	\$365.00	7.2 MP	3x	Internal Memory	32 MB	2.5 in	1.5 in	6.9 oz	choose
Canon	\$439.99	7.1 MP	3x	SD Memory Card	32 MB	2 in	1.04 in	6 oz	choose
Fuji	\$253.00	6.3 MP	4x	XD-Picture Card	16 MB	2 in	1.4 in	7.1 oz	choose
Sony	\$336.00	7.2 MP	3x	Internal Memory	32 MB	2 in	1 in	5 oz	choose
Nikon	\$304.18	7.1 MP	3x	Internal Memory	13.5 MB	2 in	1.4 in	5.3 oz	choose
Olympus	\$334.00	7.4 MP	5x	XD-Picture Card	32 MB	2.0 in	1.7 in	7.1 oz	choose
Pentax	\$238.99	5.3 MP	3x	Internal Memory	10 MB	1.8 in	0.8 in	3.7 oz	choose
Canon	\$273.18	4.0 MP	3x	SD Memory Card	16 MB	2 in	0.82 in	4.59 oz	choose
Nikon	\$329.95	5.1 MP	3x	Internal Memory	12 MB	2.5 in	0.8 in	4.2 oz	choose
Canon	\$316.18	5.3 MP	3x	SD Memory Card	16 MB	2 in	0.81 in	4.59 oz	choose
Casio	\$386.00	7.2 MP	3x	Internal Memory	8.3 MB	2.5 in	0.89 in	4.48 oz	choose
Fuji	\$309.18	6.3 MP	3x	XD-Picture Card	16 MB	2.5 in	1.1 in	5.5 oz	choose
Panasonic	\$386.00	5.0 MP	12x	SD Memory Card	16 MB	1.8 in	3.34 in	11.52 oz	choose
Konica Minolta	\$349.99	5.0 MP	12x	SD Memory Card	16 MB	2 in	3.3 in	12 oz	choose
Fuji	\$259.18	4.23 MP	10x	XD-Picture Card	16 MB	1.5 in	3.1 in	11.9 oz	choose
Olympus	\$253.00	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	9.9 oz	choose
Olympus	\$284.99	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	10.6 oz	choose
Nikon	\$259.18	4.2 MP	8.3x	Internal Memory	13.5 MB	1.8 in	2.2 in	9 oz	choose

图 13.4 基于偏好组织的评价展示方法[12]

13.3.1.6 被限制的用户控制

近期的工作中，Chen 和 Pu[9]对系统推荐和用户促进的评价方法进行了区分(见图 13.5)。根据此观点，我们集中讨论了他们提出的系统推荐复合评价，该方法没有用户参与定义。Chen



和 Pu 演示了一个用户查找与当前推荐的相机相关的、但有更高分辨率和光学变焦范围的相机的例子。假设没有与用户当前需求相符的复合评价建议，甚至没有能给他们一些信息的后续推荐选项，如更大屏幕、更大存储等。鉴于此，用户只能逐个地选择单个特征的评价，这就有可能会导致一个较长的交互周期和更低的决策准确度。他们建议可以允许用户更加灵活地定义他们自己的复合评价来对这种限制进行处理。假设是只有包含了后续推荐选项的单个评价选项会被展示给用户（如动态的单个评价）。用户动机的评价方法，让用户选择一个或多个评价来对这些种类（如复合评价）的选项进行任意组合。

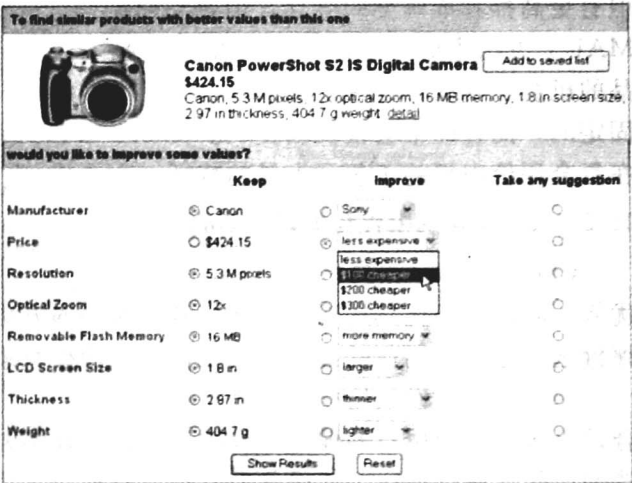


图 13.5 用户促进的评价接口[9]

Shearin 和 Lieberman [64] 提出的 Apt Decision 系统是一个早期评价系统的例子，同样实现了用户驱动的方法。在房屋租赁市场的用户可以对与推荐租房描述相关的 21 个可选特征自由地进行评判。Chen 和 Pu 选择了其他不同的方法，让用户有更大的控制能力去选用那些偏好限制来影响推荐。他们描述了用户是如何为了推荐特征的改善和折中，自主地指明其权衡标准。例如，用户能表示他们想看到一个如图 13.5 所示的推荐（比当前推荐的产品便宜一个指定的值）。

作为对比，图 13.6 显示了 Apt Decision 系统提供了用户对偏好属性内容的直接访问，用户能轻易地对偏好进行修改（增加和减少偏好程度）。除了让用户指明他们对特征偏好的重要性，Apt Decision 并不支持用户对愿意接受的妥协程度进行更加详细的指定。Chen 和 Pu 表示，通过用户促进评价的方式，用户能获得较高的选择信心和决策准确度。但是，有些用户仍然更偏好于系统提议方式的评价，他们觉得当有相关选项展示出来的情况下，这些评价的使用很直观，并且能快速引导至合适的产品。SmartClient[53, 54, 55]是设计用来帮助用户查找航班信息的系统，在偏好提取上也提供了较大的用户控制能力。它同时也作为度假套餐、保险单、住房租赁的

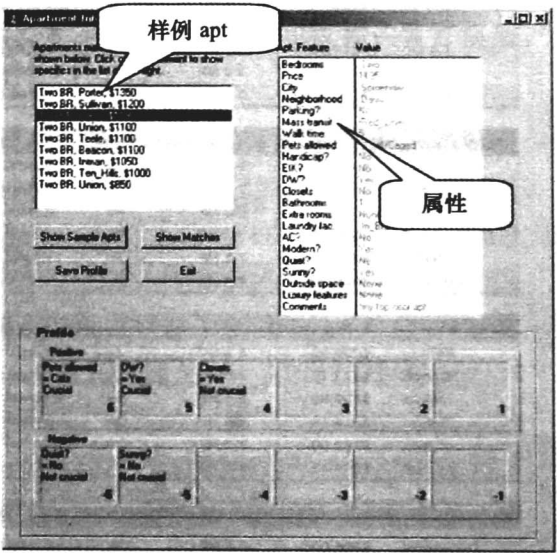


图 13.6 Apt Decision 的用户推进的评价交互截图[64]

推荐系统。产品的挑选由一个决策理论问题来表示，用户尝试着挑选更加满足他们偏好的产品。选择最好的产品通常是一个权衡的问题，因为用户需要决定在产品各种特征之中如何取舍。SmartClient将评价机制作为软限制。交互接口允许用户通过指定权重来直接为每个属性构造一个评价函数，然后推荐系统将这些包含限制的函数进行组合，为每个属性和产品计算出一个效用值。该系统展示了一个排序后的拥有最高效用值的最佳产品的列表。当用户改变了他的偏好时，不论是通过添加了另外一个评价条件，还是改变了偏好的权重，产品列表都会被重新排列。



### 13.3.2 基于评价的推荐系统中的检索挑战

与使用的评价(单个/复合)的类型或者生成评价的方法(比如,固定对应动态,系统提议与用户促进)无关,推荐是否成功很大程度上受用户评价行为的影响。如上所述,用户可能太过侧重于一个特征,导致推荐出来的产品并不满足他们所指定的其他特征[39]。然而,几乎没有什么策略来控制用户如何选择使用评价标准,为了提高推荐的效果,只有尽量减少检索的挑战。在本节,我们概述了以上一些问题,并简要地讨论了已经被解决的问题。

#### 13.3.2.1 偏好的易变性与生存期

FindMe 系统中的偏好模型是一个特征向量,从用户最初高级特征限制的实例中获得。使用一个评价会用最近特征评价来更新这个模型,并且会临时将那些不满足评价条件的餐馆从推荐候选集中删除。然后,这些经过筛选之后的餐馆会被按用户的偏好进行分层排序。

如在 13.2 节所提到的,不能指望用户在推荐会话过程中能提供一致的反馈。传统的基于评价的推荐系统(如 RentMe, Entrée)注重于当前评价和推荐,并不考虑过往所选择的评价,这同样会导致检索失败。许多用户在推荐会话刚开始时,不会对自己的需求有一个很清楚了解。这就导致用户为了绘制出一个关于可用产品的清晰图像,在会话中会选择明显不同的评价选项来探索各种不同的产品空间的领域。例如,我们可能会发现,在某个周期中,一个用户开始想要买比当前所推荐的售价为 500 元的相机便宜的数码相机,随后又要求找比另外一个推荐出的售价 500 元的相机贵的数码相机。

Apt Decision 系统[64]维护着更灵活的关于用户评价的偏好模型,该模型和其他模型的区别在于,该模型将正偏好模型和负偏好模型进行了分离,并允许用户可以对其进行访问。此系统还支持基于偏好的部分比较,以收集更多潜在的偏好信息,并添加到模型中。Shearin 和 Lieberman 认为学习一个精准的用户属性比尝试基于限制的用户偏好搜索空间的方式要更有利[64]。如果搜索空间被过度限制,用户将不会察觉到存在一些潜在选项,包含他们可能愿意折中的特征。相反,宽松的约束允许用户更多地探索产品空间,让推荐系统有机会获得更多的用户偏好,并更好地进行推荐。

SmartClient 系统也将使用评价表示用户显性偏好。两者都选用了受约束的解决方式来获得最佳的解决方案。评价形式的用户偏好被建模成 CSP 形式的约束。使用一个代理程序不断地观察用户对偏好的修改,然后改善启发模型以提高准确度。Zhang 和 Pu[73]描述了一个 MAUT 偏好模型,用于调整基于用户评价反馈的效用,然后就能检索出最大化偏好效用的推荐。相似地,Chen 和 Pu[12]描述了一个 MAUT 偏好建模方法,让用户能通过直接操作评价标准来指明他们的权衡。

Reilly 等的增量评价[58]方法,维护了一个组成真实评价的用户偏好模型,到目前为止仍被用户使用着。解释这种情况的直觉是,被沿用至今的评价代表了演变着的用户需求。为了保持用户模型的准确度,去掉了相互矛盾的评价,正如所有现存的评价中的最近评价是经过精选的。在检索推荐时,待选项的优先顺序应该满足:1)符合当前的评价;2)与之前的推荐相似;3)满足大多数之前的评价。<sup>①</sup>给定两个与之前推荐同样相似的候选项,他们的算法更倾向于选择与最近大量的选择保持一致的评价,例如,那些返回更高通

① 在这种做法中,他们显式地将用户模型中过往的评价作为软限制,用来在之后的推荐周期中进行限制;在推荐中,满足所有之前的评价并不重要,但是能满足更多的评价的更能作为推荐待选项。

用性分数的选项。结果就是，能够减少近 70% 的会话长度[58]。

最后，Nguyen 和 Ricci 最近推动了维护可替换的偏好模型的需求，用来区分长期的和特定会话的用户偏好。他们提出了一个方法来整合这两种偏好信息，用于生成个性化的推荐[41]。特定会话的偏好，包括上下文偏好与产品特征偏好。上下文偏好，比如，一个餐馆在某时间开张或位于用户附近；产品特征偏好，比如，便宜的中国饭馆。长期用户偏好[44]是指关于用户的那些相对能维持很长一段时间的信息，比如，多个连续的推荐会话。这些偏好通常在注册的时候引出，并在之后使用系统的过程中被修订。

13.3.2.2 减少的与不可达的选项

如上所述，早期基于评价的推荐系统用评价选项作为产品选项的临时过滤器，并且作为进一步的考虑，对所有被评价的推荐进行缩减。McSherry 与 Aha 发现了一个关于减少选项问题[40]的潜在缺点。他们认为，当仅有的一个可以为用户推荐的产品，且这个产品是他们已经评价过的产品，而为了防止返回他们已经评价过的产品，这个产品也会被排除，那么就会出现检索失败的情况。他们展示了一个例子(图 13.7)，如果展示给用户一个实例 1(比如，有一个在地址 A 的 3 间单独卧室的房产)。假设用户喜欢在指定地方的有 4 间单独卧室的房产。假设那些特征具有同样的权重，两个实例的相似度取决于匹配的特征数量，基于卧室特征(比如，床的数量大于某值)的评价可以预见该用户会选择实例 2。如果没有一种情况能满足用户所有的需要，他们就会重新评估实例 1，但是实例 1 已经被移除，所以他们只能考虑实例 3，或者重新开始推荐的会话。



图 13.7 显示减少选项问题，导致减少评价推荐问题的结果[40]

相反，不可达问题[40, 18]是指没有对之前的推荐进行排除。在[40]引用中，McSherry 和 Aha 也提出这种情况可能会存在导致满足用户需求的产品(如果有)始终都不会被检索出来。这个问题在图 13.8 中很好地进行了展示，将选项分好类之后按顺序推荐给用户时，第 4 个评价将用户带回了最初的推荐，而没有展示出实例 5。这里的一个重点是，该例子中的推荐检索只被最近选择的评价所影响，即最初的查询条件以及其他之前的所选择的评价并没有被考虑到。

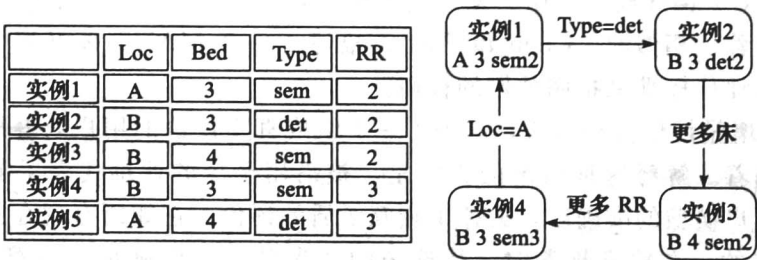


图 13.8 显示不可达问题，其导致未排除被评价的推荐[40]

McSherry 和 Aha[39]描述的 Tweak 系统实现了渐进评价方法，类似增量的评价[58]，解决了需要维护用户历史评价记录的问题。根据用户的评价检索出来的推荐项必须同时满足历史评价和最初的查询条件的限制。图 13.9 显示了一个推荐选项，满足了一组用户的当前限制，由他们初始的查询条件(比如，制造商=Dell 同时价格≤1000)和历史评价(比如，类型是便携式电脑

且屏幕尺寸小于某值)组成。注意,类型是便携式电脑这个限制条件,是用户在此处提供的仅有条件,并不符合当前的推荐。渐进评价方法检索出那些与当前推荐和评价最相似的产品,优先处理那些在偏好模型中的偏好属性。该模型为特征值记录了明确的、假设的和预测的偏好。这里明确的偏好是指那些初始查询条件,并随后在象征性的特征(例如,类型是便携式电脑)上作为替代评价应用。假设的偏好是指那些决策性的偏好,这样暗示关系的偏好特征。比如,在价格中,假设越少越好(less-is-better, LIB)可能是理由充分的;又如,内存,可以假设越多越好(more-is-better, MIB)。预测的偏好是指那些用户趋于拥有理想值的特征,并且偏好于那些接近于理想的值,如越接近越好(nearer-is-better, NIB)[37, 35]。

推荐实例	当前约束		偏好	
make=Dell	make=Dell	Y	make=Dell	E
price=719	price≤1000	Y	price=658	A
type=desktop	type=laptop	N	type=laptop	E
screen=15	screen>12	Y	screen=13.3	P
speed=1.6			speed=2.2	A
memory=512			memory=2000	A
hard disk=20			hard disk=120	A
chip=Intel Pentium				

图 13.9 展示在典型渐进评价会话中的可用知识;当前推荐、当前查询和用户偏好[39]

与增量的评价不同, Tweak 系统中的偏好模型针对某特征的 latest 评价进行修整, 与此特征上的历史评价没有关联。比如, 使用了 Sony 制造的评价来代替 Dell 制造, 就是在偏好模型中用 Sony 代替了 Dell。对 NIB 特征来说, 预测的偏好值要是最近评价的推荐项(比如, 一个小于某屏幕值的评价可能会导致屏幕的尺寸值设置为 14)。相比之下, 增量的评价导致了以下情况: 在一系列的推荐环节中, 多次为一个用户展示一些关于 Dell 的选项, 然后用户选择了这些选项(比如, 它们更优先于别的变化的评价)。增量的评价使用兼容性的标准来衡量每个候选产品与当前产品的相似度, 以及其能满足的历史评价的数量。但是, 这里潜在的一个问题是: 一个高相似度的产品会带来较多的影响, 并在多次的检索中满足较多的评价。与 Nguyen 和 Ricci[41]、Salamo 等[63]类似, 当他们将相似度作为评价推荐的第二检索标准时发现了该问题。

与 Shearin 和 Lieberman[64]一致的是, McSherry 认为用户应该可以理解当前系统对用户偏好的分析方法, 并且这对于防止所谓的渐进失败特别重要。比如, 不存在一个产品能满足所有用户的要求。在 Shearin 和 Lieberman 的 Apt Decision 系统中, 用户能够持续地、直接地访问他们的偏好模型。同时这让用户对为何会检索到如此的推荐项能够有更深层次的理解, 但是仍然会产生连串失败。McSherry 选择了一种基于解释的有趣方法, 当所有的推荐候选项都不满足偏好限制时, 为用户提供可能适合的折中方案及其解释。最近的演化评价算法[40]专注于定位不可达问题。这种演化评价实现的关键不同点在于之前推荐过(如评价方式)的物品不被删除, 这点与增量的方法一样。

### 13.3.2.3 改善推荐检索过程

即使大多数像 QwikShop、EasyShop、CritiqueShop<sup>⊖</sup>、Entrée、RentMe 等基于评价的推荐系统专注于在每个推荐周期中只展示一个候选项, 而其他类似 Apt Decision 和

⊖ <http://www.critiqueshop.com/>

MobyRek 等系统为用户提供了多个推荐项以供选择。优化推荐的候选项是一个挑战。在该领域的早期工作中,介绍了 EntréeC 这样的混合推荐系统[5, 6],将协同过滤(见第 5 章)的组件加入了 Entrée[7]系统中,创建了基于知识的/协同的串联混合。像 Entrée 一样,它使用了基于用户所表明与餐馆相关的偏好知识来进行推荐。协同过滤被用来对基于知识的推荐系统所返回的推荐项的排序规则进行优化。在这里使用串联方式的关键好处是比类似加权混合的方式更加高效,后者需要在所有的物品上进行处理,而前者的第二步只关注那些需要进行额外辨别的物品。此外,串联的方式能容忍低优先技术操作所产生的噪声数据,而高优先级推荐返回的结果只能被优化不能撤销。EntréeC 的实验表明协同过滤确实能提升仅靠基于知识的 Entrée 系统的效果。在文献[6]中,Burke 讨论了存在于可替换推荐的技术和检验之间的权衡,在一些其他传统的混合推荐系统中被使用的结合方法,也与基于评价的推荐系统有关(请见第 22 章、第 2 章和第 24 章)。

Faltings 等[15]讨论了在评价系统中关键的问题是如何设计出最好的帮助用户发现他们最喜欢的解决方案。他们提出了两个关键的需求:1)推荐的项必须激励用户表达他们将来的偏好(比如,展示出可选替换的范围);2)展示给用户的推荐项必须包含用户可能会考虑的最佳解决方案(假如,当前的偏好模型是完整的),故用户能选择用来作为最终的解决方案。动态评价的方法试图通过组合评价的方法来发现那些能引导用户的需求。但是,依靠个体评价的系统也许能找到它所提供的低特征级别的反馈,不需要充分的细节就能明确地专注于下一个推荐环节[33]。比如,推荐的相机拥有更高分辨率能帮助用户缩小搜索范围,但是仍有可能导致检索出大量的推荐候选项。评价和数值启发反馈模式(例如,为我展示 500 万像素的相机)就能更有效地降低产品选项的数量。

在与数值启发方式比较时,McGinty 和 Smyth[32, 33]论证了基于评价的推荐系统在推荐会话中会有延迟。作为一个潜在的解决方案,他们提出了一个新颖的切换策略,此策略中检索模式与用户评价行为是一致的。他们论证了自适应选择策略是如何在标准的基于相似度的方法和多样化强化检索方式的基础上减少了 60% 的潜在交互。

在最近的工作中,Chen 和 Pu[9]描述了他们的样例评价的方式,他们发现用户的偏好通常是基于上下文的。从其中展示出来的多个推荐项中,用户选择了其中之一来进行评价。他们选用了一种基于偏好的组织方式[12]来推荐最高效用的产品,同时允许用户在多个特征上自由地进行组合评价,此方法表示用户希望使用简单或者复杂度折中的选择。

#### 13.3.2.4 多用户偏好的处理

为一群用户推荐物品有很多的挑战(例如,参考[3, 47, 24],以及第 22 章、第 18 章、第 21 章和第 16 章)。Jameson 在这个领域中的早期工作[20],重点提出了一些难点(简述见图 13.10),并且描述了在一组原型推荐系统上下文中如何处理:旅游决策论坛(Travel Decision Forum)。这些系统也和单用户的推荐系统有同样的问题(比如,偏好的不一致、发散性),以及其他的关键区分特点,包括促进组成员之间偏好的相互了解并达成共识(参考 13.4 节,交流交互如何在这里扮演了一个重要的角色)。生成一些能让多用户感兴趣的推荐项集合本身就是有挑战的,更多的挑战是如何在多用户参与实时同步的推荐会话中记录并组合这些偏好[29],并处理其中的冲突。这里的关键目标问题是:1)如何管理多用户交互,并进一步从多个同时使用的用户中收集反馈和偏好?2)如何动态地维护个人和组的偏好模型,使得结果能有可能同时满足单个和组的用户?

McCarthy 等最近的工作[29, 31]集中在基于评价的组推荐架构中如何使偏好聚合并达成共识。他们介绍了协作咨询旅游系统(Collaborative Advisory Travel System,



CATS), 此系统被设计用来为一群一起要在滑雪度假计划上达成共识的朋友提供帮助。该系统通过同时使用动态单元、组合评价、单用户和多用户的偏好模型来提供单用户和多用户的反馈模型。合理的假设是: 单个用户在使用这种系统的时候, 在看了同组其他人的反馈之后, 可能需要重复查看之前的推荐项。比如, 一个用户选择了在节假日的开销上进行妥协, 那么之后其他同组的成员可能也会这样做。CATS 通过促进生成主动和被动推荐来提供以上描述的功能。被动推荐是指那些展示给单个用户的建议, 是根据个人的用户模型所产生的评价做出的相应反应。相反, 主动推荐是指由系统自动为群组生成满意度非常高的群组偏好的推荐候选项; 此处没有考虑的是, 单个的组成员可能之前拒绝过这个推荐项。另外, 组成员也能为系统辨别出哪些推荐候选项是整个群组的潜在偏好。偏好模型中的偏好不一致可以通过增量评价方法来处理。组模型在生成推荐项时可能包含一些冲突的偏好和目标, 这些不一致被最小化, 以便最大化地兼容多用户的偏好。

推荐过程阶段	来自单个成员的推荐的差异	新问题
1. 组成员指定他们的偏好	组成员可能想要查看其他每个成员的偏好说明	这个研究的优缺点是什么, 以及系统如何支持
2. 系统生成推荐	必须应用一个偏好聚合程序	聚合过程怎样有效地阻止人为操纵的偏好说明
3. 系统将推荐展示给组成员	单个成员解决方案(可能不同)的适用性成为整个解决方案的重要层面	与单个成员相关的适用性如何有效展示
4. 组成员决定接受哪个推荐(如果有)	最终决定不必由一个人决定; 需要达成共识	当组成员无法参与面对面讨论时, 系统怎样支持得出最终结论的过程

图 13.10 Jameson[20]所简述的面对组推荐架构时的挑战

据我们所知(在我写本书的时候), 并没有其他在基于评价的推荐领域中的工作是专注于多用户偏好的建模的。但为了完整性, 我们描述了与典型的(非基于评价)组推荐的比较。包括存在的多用户的协商应用, 覆盖了虚拟环境[50]和真实环境[16]。在大部分这些系统中假设了一个自动协商机制, 它以系统中的现有的静态个体偏好模型为基础。CATS 系统的实时交互特性不适合使用静态的偏好模型。在 CATS 中, 个体偏好模型很容易受多用户反馈的影响而趋于多变。其他组推荐领域的研究中包括 MusicFX 系统[25]。MusicFX 是一个组偏好模型推荐系统, 自适应选择播放的音乐来最好地满足那些处于相同环境中的群组的音乐品味。PolyLens[46]是 MovieLens 的一个泛化版本, 其通过偏好建模来给群组用户推荐电影。相反地, 它还提取了相似上下文中的用户历史音乐偏好记录来进行协同过滤。类似于 CATS, 旅游决策系统(Travel Decision Forum)[20]帮助一组用户在共同的假期出游上达成一致。但是, 它致力于帮助那些不在同一个地方的用户。在其他相关的工作中, Plua 和 Jameson[49]提出了一个方法, 使得那些用户即使对相关领域知识了解不太全面, 也能从其他同组用户那里获得帮助。与 CATS 不同, 这个系统为一个群组提供异步的交互, 而不是同步的交互。

### 13.4 评价平台中的交互研究

在根据不同领域和平台的特征为其设计推荐的交互接口时, 设计者面临着具有差异的技术和适用性的挑战。本节专注于在现有的基于评价的推荐系统中如何设计决策系统的(见第 16 章)。毫无疑问, 一个共同的问题是如何能将用户对系统的认知透明度和操控深度权衡好, 保证最小化用户认知程度和交互负担。



### 13.4.1 扩展到其他评价平台

即使当前大部分基于评价的推荐系统是基于网页的桌面平台(如 FindMe 系统、QuickShop、EasyShop、CritiqueShop、AptDecision、SmartClient),评价策略在其他平台中也可以被论证为一种有效的交互方法。比如, Ricci 和 Nguyen[61, 43, 62]专注于移动设备上的推荐交互设计和评估。他们指出只有很少为移动用户设计的基于网页的推荐系统,而且这些推荐系统没有一个具有交互功能。持久的直接操作在这里是不实际的。关键的挑战包括:这种设备只有很小的屏幕尺寸,比传统的 PDA 设备(比如, Palm/奔迈或掌上电脑)拥有更少的键盘功能,并且也只有有限的计算能力。他们的 MobyRek 移动交互界面以及基于网页的推荐系统用于帮助用户提前制订旅游计划。重要的是, MobyRek 能为已经上路甚至已经到达了目的地的用户提供及时的支持。

在移动的领域中,图 13.11 显示了 CritiqueShop 的交互接口是如何扩展到 iPhone 上的。影响设计的主要考虑因素包括有限的屏幕区域和更多用户直接操作机会(记住 iPhone 有一个触摸的界面)。这里提出了另外一个设计方法,其移除了基于文字的组评价的展示方式,而转向展示更多可视的替代选项(之后会进行讨论)。



图 13.11 CritiqueShop 的 iPhone 可视界面截图[www.critiqueshop.com]

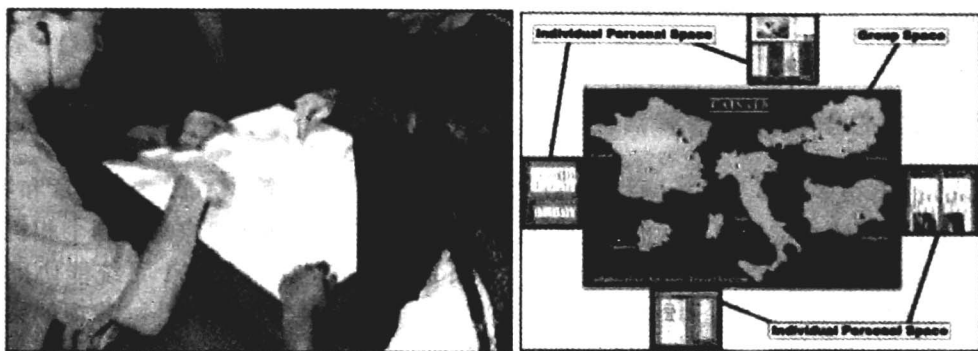


图 13.12 演示 DiamondTouch 系统的 CATS 互动

在其他的工作中，McCarthy 等[30]论述了 CATS 的群组推荐系统是如何在多用户 MERL 的 DiamondTouch<sup>⊖</sup>平板设备上交互操作的。DiamondTouch 平板是一个多用户的触摸-手势触发的屏幕，为一个组提供协作操作。用户同时与界面进行交互，浏览潜在的滑雪度假选项和返回的评价推荐(比如，之前提到的推荐项不断受到个体变化的兴趣和其他群组成员的评价模式的影响)。此处用户还能对其他用户展示潜在感兴趣的选项，就像根据自己的偏好面对面地为他们提供推荐，只是使用了不寻常的交互行为。

#### 13.4.2 用户直接操作与限制用户控制的比较

影响评价交互方式设计的一个基本考虑因素是，在启发精准的偏好信息以及获取到该信息所需的相关代价之间找到平衡点。早期的基于评价的推荐系统提供了非常严格的用户控制方法(比如，展示了静态的评价，并且用户只能选择其中之一)。之后的系统，如 QwikShop、EasyShop 和 CritiqueShop，通过可替换(单个/组合)的评价选择模式为用户提供了更多的自助可控性，但不允许用户直接修改他们的偏好模型(如 Apt Decision)，或者在确切的特征上设置明确约束的边界(如类似 SmartClient 的情况)。

Chen 和 Pu[11]演示了他们的混合交互界面是如何同时支持用户驱动和系统驱动的评价，并且让用户能达到更高的决策准确度和交互满意度，同时比系统建议的评价交互方式需要更少的认知。为了替代计算出推荐项之后让用户选择的方式，自驱动的评价方式专注于展示样例和激励用户按照他们自己所希望的那样对任意的特征进行组合，从而进行单独的或组合的评价选择。重要的是，在关于个体特征的改进和妥协上，用户还应能自由地描述他们的折中标准，并且看到更加接近他们理想选择的一组新产品(见图 13.5)。McSherry[38]在进化的评价推荐系统中寻找一种折中方案，除了系统生成的松弛机制，可以引入直接的概念和源于用户动机的松弛评价。Tweak2 系统实现了这个混合主动的方法，允许用户请求一个像当前推荐项的物品，但是在他们选择的特殊(之前评价过的)特征上没有对选项的值进行限制。

移动平台为用户提供了非常有限的控制许可[42]。此外，电信服务的花销倾向于不鼓励用户参与较长时间交互的会话。在这里，通常用点击次数、翻页次数、键盘操作次数和用户需要进行操作的次数来衡量效率。由于要保持低的消耗(时间和金钱)，往往需要牺牲用户的可操作性。在 MobyRek 系统中，影响评价是否被采纳为合适的直接反馈机制有两个关键的要求：1)需要在非常小的屏幕区域中获得有用的偏好信息；2)用户系统的交互必

⊖ DiamondTouch 的产品线从 MERL 研究实验室移到了名为 Circle Twelve 的一个独立公司。

须是低消耗的；就是说需要在最短的时间内获得有用的推荐。评估显示通常会在 2~3 个周期/点击之内[42, 61]为用户推荐可以接受的选项<sup>①</sup>。

### 13.4.3 支持性解释、置信和信任

本手册中有很多章节与该标题相关(以第 20 章、第 14 章、第 15 章、第 16 章和第 25 章为例)。推荐系统建立与用户之间的信任并能用推荐项来说服他们(就像购买哪个相机、个人电脑)的能力是一个至关重要的设计因素[51, 52]。如果用户意识到这个系统能协助他们做决定,那么他们就更有可能会再次使用这个系统。研究者通过为用户解释推荐项的来源,实现在基于评价的推荐系统中建立信任的设计原则和策略(例如, [52, 57, 66])。通常,推荐一个产品推荐系统可能要解释: 1) 为什么这个特殊的产品会被推荐(或不被推荐)[36]; 2) 存在什么样的时机: “我可以从这里到哪里去”, 什么时候会展示不合适的推荐项[57]。在第一个例子中, 为什么产品会被推荐出来, 能够很容易地通过为用户显示用户模型中的信息来解释, 正如 Apt Decision 系统中合理的解释方法[66](见图 13.6)。McSherry[36, 37]强调当推荐结果不满足用户需求时, 解释检索失败原因的重要性。他描述了一个混合主动的方法, 通过对不满足查询条件的检索的特征进行突出标记(比如, 没有低于 300 欧元且分辨率高于 400 万像素的相机), 让用户能够修改他们的限制条件, 以便从检索失败的情况下进行恢复。



图 13.13 MobyRek 移动端餐馆推荐系统[61]

Reilly 等[57]讨论了动态的组合评价能帮助用户对各种特征之间的共同交互进行辨别(比如, 通过提升系统的透明性来进行解释[66]), 从而更好地理解存在于当前推荐环节的推荐时机。在一些推荐领域中, 用户通常都不会对各种特征的细节有比较完整的理解, 所以组合评价能够帮助更有效地标出这个空间, 并将决策的错误降到最低。比如, 在数码相机领域中的标准评价中, 用户可能很简单地选择小于某价格的单个评价, 并错误地认为能得到一些比所指定的价格低, 且满足他们其他所有要求的相机。但是, 用这种方式来降低价格可能会导致检索到一些分辨率不能满足用户需求的相机, 结果就是, 用户需要返回去重新进行检索。Hadzic 和 O'Sullivan[18]强调了一个更深层的潜在问题, 评价的不对称对用户来说可能有悖常理。所以, 想要撤销一个评价并不能单靠点击浏览器的回退按钮来实现。但是这种问题在组合评价(价格小

① 目的是评估一个移动平台的餐馆应用, 使用蓝牙连接的 GPS 接收器来收集位置信息。

于,分辨率小于)中并不容易出现,因为用户在选择评价之前会对价格下降所产生的影响有所了解。另外,QwikShop在界面上预留了一个区域来提供额外的解释。用户还能得到一些信息:与每个展示的评价选项相关的产品数,每个选项的特征的值的范围,见图13.2。

通过解释来提供系统的透明度,让用户更加有把握地去评估系统的可靠性(比如,提升用户的信任)。系统置信度是对解释的一种补充。为了在推荐中最大化用户的信任,系统本身就应该能够在推荐中评估自有的置信度(或缺少置信度)。Reilly等[59]提出了一个系统级别的对置信度进行建模的方法,用于基于评价的推荐系统中。用户通过推荐系统在推荐过程中的置信度,来判断是否相信推荐结果。他们提出了一个由反馈来影响的模型,该模型在特征和产品两个级别上对系统的置信度标准进行了计算。例如,一个在价格特征上较低的系统置信度分数,可以理解为当前推荐的重心集中在为用户提供正确的价格范围时,系统会不确定其推荐的物品是否是可信的。一旦用户对这些特征有了更清楚的理解,他们可能更倾向于重新调整他们的反馈(比如,优化和改进他们的偏好模型)。为了指导推荐系统能够做出更加让人信任的建议,作者还演示了产品级别的置信度分数是如何对现存的相似度知识进行补充的。

#### 13.4.4 可视化、自适应性和分区动态性

第17章强调了在产品推荐系统中使用可视化界面的好处,本节专注基于评价的产品推荐系统。Zhang等[72]介绍了一个使用有意义的图标来代替文字表示的组合评价的可视化界面。图13.14展示了一个简单的组合评价是如何使用这两种方式来进行展示的。其研究表明,用户更乐于选择可视化的组合评价,而不是文字形式的组合评价(可视化方式的选项被选择的频率比文字的增加大约50%),并且随后的好处是还降低了交互的时间(会话时长降低大约53%)<sup>①</sup>。图13.11显示了Zhang等是如何通过操作iPhone提出了可视化的界面。

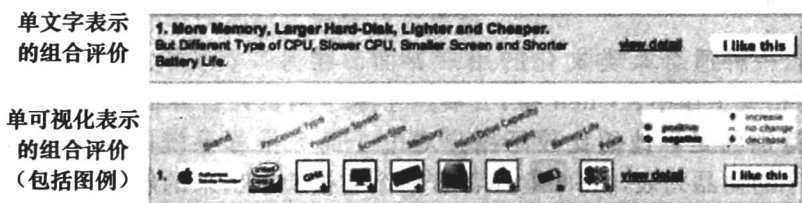


图 13.14 演示了简单的组合评价文字和可视化表示方式[72]

其他研究显示了,支持根据用户评价的反馈来动态改变界面图标的自适应的可视化界面的威力。特别提到的是,Averjanova等[2]描述在MobyRek推荐系统[61]中,通过基于地图的(map-based)可视化界面,能将推荐效率(比如,平均会话时长减少17%)和用户满意度提高。类似地,CATS群组推荐系统的可用性研究显示了动态自适应界面[30],促进了改变多用户偏好时彼此间的互相感知。两个系统共有的一个关键的设计性考虑是如何最好地定位推荐结果上受限的评价影响的问题,这个问题导致了用户很难看到评价的效果。在上面提到的两个系统中,该问题是通过真实用户的可用性评估被发现的,虽然用的是动态界面的组件(比如,改变图标大小和颜色编码的推荐方式),但是在两个系统中定位到该问题的方式却非常相似。例如,在CATS中,如果一个特别的常去的度假地点的兴趣级别在一个群组中都非常高,那么这个度假地点的显示尺寸就比其他低兴趣级别的要大。MobyRek界面早期的用户报告了另外一个所关心的问题是对距离的察觉,对用户来说,

① 在操作笔记本电脑数据集上,用一个版本的系统收集数据的结果。

在移动中比较餐馆和当前用户的位置非常难。在修改的 MapMobyRek 系统中进一步的扩展包括一个基于地图的界面用来处理这个问题。其他的扩展包括颜色编码, 用来展示推荐的适合程度, 以及提供物品逐个比较的功能。

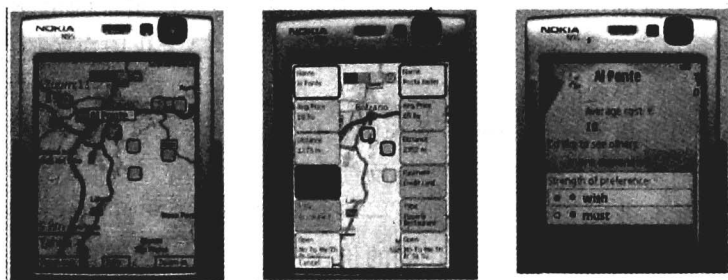


图 13.15 演示了 MapMobyRek 移动端推荐系统的可视化界面

类似于 MobyRek, CATS 实现了一个基于地图的界面, 但是一个额外的设计考虑是如何最好地表达所有组成员达成一致的进度。该功能可以通过使用高可视化的颜色编码的一致性指示器很容易地达到, 该指示器通过候选度假选项的比较来简述演化的用户建议。在设计自适应的界面时需要额外考虑的是承担用户对在会话中变化太快的困惑的风险。QwikShop、EasyShop、CritiqueShop 对他们的交互界面[69]中动态和静态的元素进行了划分。这也是 CATS 推荐系统中的一个关键性的设计考虑, 在这里很有必要让用户保持对其他用户的偏好和动向的关心, 并不会产生困惑。这个是通过使用直观且确切、共享且独立的空间和精心设计的可视化指示来进行管理的。另外, 介绍了一些动态界面的组件来进行群组偏好进展的信息交流和监控达成群组一致的进度。

#### 13.4.5 关于多文化的适用性的差异

研究表明, 一种适合某类文化的用户界面, 在另一类文化中却并不适用[21, 23, 45]。在最近的相关工作中, Chen 和 Pu 展示了基于网页评价界面[13]的一个全面交叉文化评估方式。特别的是, 他们比较了用户对电子商务推荐系统的两种展示策略的反应: 1) 作为一个排序的物品列表, 每个物品都有一个它被检索出来的解释; 2) 作为一个基于偏好的组织, 为了许多群组的推荐系统共同的差异/权衡与排在前面的产品相关, 对这些推荐系统进行了分类和概述。非常简要的是, 120 个参与者(60 位西方人, 60 位东方人)的主观评估显示了在他们的理解推荐质量和他们的总体满意度上, 有组织的观点影响了大多数人。想了解关于主观观点的更加全面细致的比较请参见文献[13]。

### 13.5 评价的评估: 资源、方法和标准

在本节, 我们最开始指出了一些资源, 这些资源通常在这个领域中被用来作为评估的目标, 并且提供了一些它们可见的细节。接着, 简述典型的评估方法, 并列出了一些经常被使用的关键评估标准。

#### 13.5.1 资源和方法

那些刚刚在这个领域中起步的人需要知道, 在学术出版物中有很大量被用来作为评估目标的数据集可以免费下载<sup>①</sup>。比如, Travel(在[26, 58, 34]使用)、PC(在[58, 39,

① <http://cbrwiki.fdi.ucm.es/wiki/index.php/CaseBases>



40, 34]使用)、Whiskey(在[34]中使用), 还有 Digital Camera(在[27]中使用)等这些数据集。Entrée 数据也可以免费获取<sup>①</sup>。在其他的工作中, 还使用了公寓[55, 73]和滑雪度假[29]数据集, 但是这些不是公开的。

总体来说, 这里采用了推荐系统所通用的大多数的评估方法(见第 8 章和第 15 章)。在理想的情况下, 当评估基于评价的推荐系统的性能(主观和客观)时, 实时的适用性研究和性能评估必须最终由真实的用户来进行。但是, 招募到足够数量参与多次测试的志愿者却是比较困难的。这里的一个解决方案, 通常会在一些不同的数据集(上面提到的)和人工生成的用户属性来进行模拟的研究(在随后的真实用户的情况下, 产生的结果期待被验证)。如果设计得足够好, 这些离线的模拟可以确信用来显示真实环境中的性能。在基于评价研究的文献中通常被广泛采用的一个进行性能评估的方法, 是留一验证方法[33]。该方法非常简洁, 通过使用这个方法, 数据集中的产品被设置为推荐目标, 并且被暂时从数据集中移除。它的特征的一个子集被选择作为初始的查询条件。在每个推荐的环节中, 如果评价被选择, 那么它们就与推荐目标的特征一致。在推荐出与理想的产品最相似的产品之后, 模拟的推荐会话就结束了。重要的是, 实验性的装置往往会有一个相关的控制装置(比如, 使用一个替换的检索策略或者反馈方法), 用来允许对结果的重要性进行合适的评估。遗憾的是, 唯一可以接受的可用性评估的方法, 却是通过真实用户的交互实验。

### 13.5.2 评估标准

在评价研究的评估中使用的共同客观性能衡量办法包括:

**效率:** 明确地用用户在找到他们的目标产品前[39, 40, 73, 38, 72, 13, 61, 58, 60, 2, 27], 所参与的会话长度的减少来表示(比如, 评价环节/交互的次数的减少), 找到一个推荐目标项所花的时间[10, 13, 2, 68]。

- **准确度:** 有很多种途径来对正确性进行衡量。包括:

- **评价预测准确度:** 指展示出来的评价匹配的次数, 即被真实的用户选择的次数[12, 60, 6]。

- **评价应用频率:** 指用户选择的评价的时间的比例(比如, 用户偏好模型的相关性)[73, 72, 60, 11, 27]。

- **推荐准确度:** 由 Chen 和 Pu 在[12]中作为所选择的评价的推荐周期的百分比进行衡量, 在这个推荐周期中展示了推荐目标(即推荐目标在被推荐的产品中), 并且在[61, 40, 37, 68]中作为成功会话的百分比。

- **决策准确度:** 指时间的比例, 这些时间中最终被用户接受的推荐是最好的解决方案。在[9, 11]中作为衡量用户在看到所有可替换推荐项时改变他们想法的百分比。在[44]中作为衡量位置替换的度。

- **交互代价:** 往往指平均会话点击距离, 或者达到指定目标所需的交互(如评价)次数。假设在各推荐周期中被选择的一个评价, 等同于衡量会话长度[12, 10, 13, 11]。

评价研究中的真实用户的适用性性能评估, 通常调查了参与者和探寻他们在如下 Likert 度量表(Likert)主观反馈[9, 10, 62, 61, 13, 60, 26, 11, 2]。

#### 系统设计

系统透明性: 用户是否了解为什么做推荐系统?

① URL-<http://kdd.ics.uci.edu/database/entree.data.html>

用户控制:	用户是否能感觉到在交互中, 他们能控制自己的偏好设置?
能力	
意识到易用性:	用户能否发现系统很容易使用?
意识到有用:	用户能否感觉到系统是很有用的(如相关的推荐、好的解释等)?
决策置信度:	用户觉得“最好”的产品有多么可信?
意识到努力:	用户多容易就能获得他们正在找的信息?
意识到准确度:	建议的准确度的程度?
可信度	
满意度:	用户对交互的满意度是多少?
推荐的可信度:	用户能否感觉到推荐系统一致地提供与他们的偏好相适应的建议?
用户意图	
购买意图:	用户是否愿意在某些情况下购买推荐的产品?
返回意图:	用户是否会(一次次地)返回来重复使用系统?

### 13.6 总结与展望

在本章中, 我们展示了过去 10 年中, 评价研究进展的一个非技术性的概述。我们认为这可以为刚进入这个领域中的研究者提供一个有用的参考。我们描述了在现存的评价系统中应用这种方式的好处。学术界所关注的关键性问题和挑战, 表现为能在如下方面取得多少进展: 1) 鉴于优化偏好获取的自动的评价展示; 2) 改进的基于评价的推荐系统的检索性能。另外, 我们概述了一些设计考虑, 它们影响了跨平台和用户环境反馈的评价模式。最后, 我们简述了一些侧重于评估基于评价的推荐系统的热门评估标准。

尽管在评价的领域中有相当多的研究活动, 但仍然存有很多挑战和机遇。这里所能提到的远远不够, 但是为了对这章进行总结, 我们对一些例子进行了强调。首先, 大部分现存的系统都假设了用户会进行正向评价, 用以表示偏好某推荐的物品(比如, 为我展示像物品 A, 但是更便宜的物品)。有些时候, 用户选用一个反向的评价是有意义的, 就如“不要再为我展示像 X, 并且位于 Y 的物品”, 这些是有争议的, 正如了解用户的需求和偏好对系统来说很重要一样。

在入门决策话题上的相关工作没有太多, 这些工作集中于将什么推荐物品放在第一[15, 41]。最近, Hadzic 和 O'Sullivan 的工作中, 介绍了评价图的概念, 此概念作为一个关于用给定推荐的评价可以到达的产品集的正式推理基础。他们提出了一个有用的基础计算最佳入门推荐, 即使用最小目录封面的概念(比如, 分辨推荐候选集, 如果评价是最优的, 那么从这个集合中可以到达其他的推荐)。重要的是需要注意不是从给定的推荐候选项可以到达所有的推荐。在相似的分支中, 作者描述了如何确定存在没有被其他产品所覆盖的推荐(例如, 通过应用任何的评价都不能够到达)。在这种情况下, 一个有用的原则可能会优先考虑加入这些条目。这个想法还没有被现有的研究所实现, 或者更深层地挖掘出来, 尽管它给出了非常美好的前景。

从评价的反馈中收集用户偏好的建模挑战, 仍然是当前研究所持续发掘的一个主题。其他用户偏好建模的方式和现有方式的比较评估也非常有趣。并且, 近期的工作中也在关注维护分开的长期和短期的偏好模型[44]的想法, 但是没有工作是注重于信息是如何在它们之间被转换和维护的, 并且如此的做法可能会有什么好处或结果。对多用户偏好聚合的更进一步的方式的调查是另外一个开放的挑战, 迄今为止在这个领域中只开展了很少的研究工作。早期在基于评价的群组推荐系统[29]的研究工作中, 展示了一个协同环境中这种

方式的好处,假定是同步的反馈,并且还存在其他的机会。例子包括,其他对异步的评价的扩展,且/或非协同的环境(在这里,可能多个用户没有用协作的方式来进行交互)。这种研究可能导致在应用领域中采纳评价的方式作为交互的模式,这种评价的方式还未被当前的评价研究所发掘,例如,做游戏的方式。

## 参考文献

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I.: Fast discovery of association rules in large databases. *Advances in Knowledge Discovery and Data Mining* pp. 307–328 (1996)
2. Averjanova, O., Ricci, F., Nguyen, Q.N.: Map-based interaction with a conversational mobile recommender system. In: *UBICOMM '08: Proceedings of the 2008 The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pp. 212–218. IEEE Computer Society, Washington, DC, USA (2008)
3. Baatarjav, E.A., Phithakkitnukoon, S., Dantu, R.: Group recommendation system for facebook. In: *OTM '08: Proceedings of the OTM Confederated International Workshops and Posters on On the Move to Meaningful Internet Systems*, pp. 211–219. Springer-Verlag, Berlin, Heidelberg (2008)
4. Burke, R.: The wasabi personal shopper: A case-based recommender system. In: *Proceedings of the 11th National Conference on Innovative Applications of Artificial Intelligence*, pp. 844–849. AAAI Press (1999). Menlo Park, CA, USA
5. Burke, R.: A case-based reasoning approach to collaborative filtering. In: E. Blanzieri, L. Portinale (eds.) *Proceedings of the Fifth European Conference on Case-Based Reasoning, EWCBR '00*, pp. 370–379. Springer (2000). Trento, Italy
6. Burke, R.: Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* 12(4), 331–370 (2002)
7. Burke, R., Hammond, K., Young, B.: Knowledge-based navigation of complex information spaces. In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pp. 462–468. AAAI Press/MIT Press (1996). Portland, OR
8. Burke, R.D., Hammond, K.J., Young, B.C.: The findme approach to assisted browsing. *IEEE Expert: Intelligent Systems and Their Applications* 12(4), 32–40 (1997)
9. Chen, L., Pu, P.: Evaluating critiquing-based recommender agents. In: *In Proc. AAAI 2006*, pp. 157–162 (2006)
10. Chen, L., Pu, P.: The evaluation of a hybrid critiquing system with preference-based recommendations organization. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 169–172. ACM, New York, NY, USA (2007)
11. Chen, L., Pu, P.: Hybrid critiquing-based recommender systems. In: *IUI '07: Proceedings of the 12th international conference on Intelligent user interfaces*, pp. 22–31. ACM, New York, NY, USA (2007)
12. Chen, L., Pu, P.: Preference-Based Organization Interfaces: Aiding User Critiques in Recommender Systems, pp. 77–86. Springer-Verlag, Berlin, Heidelberg (2007)
13. Chen, L., Pu, P.: A cross-cultural user evaluation of product recommender interfaces. In: *RecSys '08: Proceedings of the 2008 ACM Conference on Recommender systems*, pp. 75–82. ACM, New York, NY, USA (2008)
14. Cohen S., Rokach L., Maimon O., Decision Tree Instance Space Decomposition with Grouped Gain-Ratio, *Information Science*, Volume 177, Issue 17, pp. 3592–3612 (2007)
15. Faltings, B., Pu, P., Torrens, M., Viappiani, P.: Designing example-critiquing interaction. In: *IUI '04: Proceedings of the 9th international conference on Intelligent user interfaces*, pp. 22–29. ACM, New York, NY, USA (2004)
16. Faratin, P., Sierra, C., Jennings, N.: Using similarity criteria to make issue trade-offs in automated negotiations. *Artificial Intelligence* 142(2), 205–237 (2002)
17. Ha, V., Haddawy, P.: Problem-focused incremental elicitation of multi-attribute utility models. In: *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence, (UAI-97)*, pp. 215–222. Morgan Kaufmann (1997). URL [citeseer.ist.psu.edu/ha97problemfocused.html](http://citeseer.ist.psu.edu/ha97problemfocused.html). San Francisco
18. Hadzic, T., O'Sullivan, B.: Critique graphs for catalogue navigation. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 115–122. ACM, New York, NY, USA (2008)
19. Hammond, K., Burke, R., Schmitt, K.: A case-based approach to knowledge navigation. In:

- D. Leake (ed.) *Case-Based Reasoning Experiences, Lessons and Future Directions.*, pp. 125–136. AAAI Press (1996)
20. Jameson, A.: More than the sum of its members: Challenges for group recommender systems. In: *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, pp. 48–54. ACM, New York, NY, USA (2004)
  21. Lee, K., Joshi, K., McIvor, R.: Understanding multicultural differences in online satisfaction. In: *SIGMIS-CPR '07: Proceedings of the 2007 ACM SIGMIS CPR conference on Computer personnel research*, pp. 209–212. ACM, New York, NY, USA (2007)
  22. Linden, G., Hanks, S., Lesh, N.: Interactive assessment of user preference models: The automated travel assistant. In: C.P.A. Jameson, C. Tasso (eds.) *User Modeling: Proceedings of the Sixth International Conference*, pp. 67–78. Springer Wien (1997)
  23. Lodge, C.: The impact of culture on usability: Designing usable products for the international user. In: N.M. Aykin (ed.) *Usability and Internationalization. HCI and Culture, Second International Conference on Usability and Internationalization, UI-HCII 2007, Held as Part of HCI International 2007, Beijing, China, July 22–27, 2007, Proceedings, Part I*, pp. 365–368. Springer (2007)
  24. Masthoff, J., Gatt, A.: In pursuit of satisfaction and the prevention of embarrassment: Affective state in group recommender systems. *User Modeling and User-Adapted Interaction* 16(3–4), 281–319 (2006)
  25. McCarthy, J., Anagnost, T.: Musicfx: An arbiter of group preferences for computer supported collaborative workouts. In: *Proc. of Conference on Computer Supported Cooperative Work*, pp. 363–372 (1998)
  26. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: On the dynamic generation of compound critiques in conversational recommender systems. In: *Adaptive Hypermedia and Adaptive Web-Based Systems, Third International Conference, AH 2004, Eindhoven, The Netherlands, August 23–26, 2004, Proceedings, Lecture Notes in Computer Science*, vol. 3137, pp. 176–184. Springer (2004)
  27. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: Experiments in dynamic critiquing. In: *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, pp. 175–182. ACM, New York, NY, USA (2005)
  28. McCarthy, K., Reilly, J., Smyth, B., McGinty, L.: Generating diverse compound critiques. *Artif. Intell. Rev.* 24(3–4), 339–357 (2005)
  29. McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P.: Cats: A synchronous approach to collaborative group recommendation. In: G. Sutcliffe, R. Goebel (eds.) *Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference, Melbourne Beach, Florida, USA, May 11–13, 2006*, pp. 86–91. AAAI Press (2006)
  30. McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P.: Group recommender systems: A critiquing based approach. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 267–269. ACM, New York, NY, USA (2006)
  31. McCarthy, K., Salamó, M., McGinty, L., Smyth, B.: The needs of the many: A case-based group recommender system. In: *ICCBR '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 196–210. Springer LNCS (2006)
  32. McGinty, L., Smyth, B.: The role of diversity in conversational systems. In: D. Bridge, K. Ashley (eds.) *Proceedings of the Fifth International Conference on Case-Based Reasoning (ICCBR-03)*. Springer (2003). Troindheim, Norway.
  33. McGinty, L., Smyth, B.: Tweaking critiquing. In: *Proceedings of the Workshop on Personalization and Web Techniques at the International Joint Conference on Artificial Intelligence (IJCAI-03)*. Morgan-Kaufmann (2003). Acapulco, Mexico
  34. McGinty, L., Smyth, B.: Adaptive selection: An analysis of critiquing and preference-based feedback in conversational recommender systems. *Int. J. Electron. Commerce* 11(2), 35–57 (2006)
  35. McSherry, D.: Similarity and compromise. In: K. Ashley, D. Bridge (eds.) *Case-Based Reasoning Research and Development. LNAI Vol.2689*, pp. 291–305. Springer (2003)
  36. McSherry, D.: Explaining the pros and cons of conclusions in cbr. In: P. Funk, P.A. González-Calero (eds.) *Advances in Case-Based Reasoning, 7th European Conference, ECCBR 2004, Madrid, Spain, August 30 - September 2, 2004, Proceedings*, pp. 317–330. Springer (2004)
  37. McSherry, D.: Retrieval failure and recovery in recommender systems. *Artif. Intell. Rev.* 24(3–4), 319–338 (2005)
  38. McSherry, D., Aha, D.: Mixed-initiative relaxation of constraints in critiquing dialogues. In: *ICCBR '07: Proceedings of the 7th international conference on Case-Based Reasoning*, pp. 107–121. Springer-Verlag, Berlin, Heidelberg (2007)

39. McSherry, D., Aha, D.W.: Avoiding long and fruitless dialogues in critiquing. In: M. Bramer, F. Coenen, A. Tuson (eds.) *Research and Development in Intelligent Systems XXIII. BCS Conference Series*, pp. 173–186. Springer, London (2006)
40. McSherry, D., Aha, D.W.: The ins and outs of critiquing. In: M.M. Veloso (ed.) *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, Hyderabad, India, January 6–12, 2007, pp. 962–967 (2007)
41. Nguyen, Q., Ricci, F.: User preferences initialization and integration in critique-based mobile recommender systems. In: *In Proc. 5th Int'l Workshop Artificial Intelligence in Mobile Systems (AIMS 04)*, p. pp. 7178 (2004)
42. Nguyen, Q., Ricci, F., Cavada, D.: Critique-based recommendations for mobile users: Gui design and evaluation. In: *In Proceedings of the 3rd International Workshop on HCI in Mobile Guides*, in conjunction with the 6th International Conference on Mobile Human-Computer Interaction (2004). Glasgow, Scotland.
43. Nguyen, Q.N., Ricci, F.: Replaying live-user interactions in the off-line evaluation of critique-based mobile recommendations. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 81–88. ACM, New York, NY, USA (2007)
44. Nguyen, Q.N., Ricci, F.: Long-term and session-specific user preferences in a mobile recommender system. In: *IUI '08: Proceedings of the 13th international conference on Intelligent user interfaces*, pp. 381–384. ACM, New York, NY, USA (2008)
45. Noiwan, J., Norcio, A.F.: Cultural differences on attention and perceived usability: Investigating color combinations of animated graphics. *Int. J. Hum.-Comput. Stud.* **64**(2), 103–122 (2006)
46. O'Connor, M., Cosley, D., Konstan, J., Riedl, J.: Polylens: A recommender system for groups of users. In: *Proc. of European Conference on Computer-Supported Cooperative Work*, pp. 199–218 (2001)
47. Park, Y.J., Chang, K.N.: Individual and group behavior-based customer profile model for personalized product recommendation. *Expert Syst. Appl.* **36**(2), 1932–1939 (2009)
48. Payne, J., Bettman, J., Johnson, E.: *The Adaptive Decision Maker*. Cambridge University Press (1993)
49. Plua, C., Jameson, A.: Collaborative preference elicitation in a group travel recommender system. In: *Proceedings of the AH 2002 Workshop on Recommendation and Personalization in eCommerce*, pp. 148–154. Malaga, Spain (2002)
50. Prada, R., Paiva, A.: Believable groups of synthetic characters. In: *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pp. 37–43. The Netherlands (2005)
51. Pu, P., Chen, L.: Trust building with explanation interfaces. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 93–100. ACM, New York, NY, USA (2006)
52. Pu, P., Chen, L.: Trust-inspiring explanation interfaces for recommender systems. *Know.-Based Syst.* **20**(6), 542–556 (2007)
53. Pu, P., Faltings, B.: Personalized navigation of heterogeneous product spaces using smart-client. In: *IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces*, pp. 212–213. ACM, New York, NY, USA (2002)
54. Pu, P., Faltings, B.: Decision tradeoff using example-critiquing and constraint programming. *Constraints* **9**(4), 289–310 (2004)
55. Pu, P., Z., H., Kumar, P.: Evaluating example-based search tools. In: *EC '04: Proceedings of the 5th ACM conference on Electronic commerce*, pp. 208–217. ACM, New York, NY, USA (2004)
56. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing. In: P. Funk, P.A. González-Calero (eds.) *Advances in Case-Based Reasoning, 7th European Conference, EC-CBR 2004, Madrid, Spain, August 30 - September 2, 2004, Proceedings, Lecture Notes in Computer Science*, vol. 3155, pp. 763–777. Springer (2004)
57. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Explaining compound critiques. *Artif. Intell. Rev.* **24**(2), 199–220 (2005)
58. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Incremental critiquing. *Knowl.-Based Syst.* **18**(4–5), 143–151 (2005)
59. Reilly, J., Smyth, B., McGinty, L., McCarthy, K.: Critiquing with confidence. In: H. Muñoz-Avila, F. Ricci (eds.) *Case-Based Reasoning, Research and Development, 6th International Conference, on Case-Based Reasoning, ICCBR 2005, Chicago, IL, USA, August 23–26, 2005, Proceedings, Lecture Notes in Computer Science*, vol. 3620, pp. 436–450. Springer (2005)



60. Reilly, J., Zhang, J., McGinty, L., Pu, P., Smyth, B.: A comparison of two compound critiquing systems. In: *IUI '07: Proceedings of the 12th international conference on Intelligent user interfaces*, pp. 317–320. ACM, New York, NY, USA (2007)
61. Ricci, F., Nguyen, Q.N.: Critique-based mobile recommender systems. *GAI Journal, GAI Press Volume 24, Number 4* (2004)
62. Ricci, F., Nguyen, Q.N.: Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems* **22**(3), 22–29 (2007)
63. Salamo, M., Reilly, J., McGinty, L., Smyth, B.: Improving incremental critiquing. In: *Proceedings of the 16th Conference on Artificial Intelligence and Cognitive Science (AICS05)*, pp. 379–388 (2005)
64. Shearin, S., Lieberman, H.: Intelligent profiling by example. In: *IUI '01: Proceedings of the 6th international conference on Intelligent user interfaces*, pp. 145–151. ACM, New York, NY, USA (2001)
65. Smyth, B., McGinty, L.: Improving the Performance of Recommender Systems that Use Critiquing, chap. *Intelligent Techniques for Web Personalization*, pp. 114–132. 978-3-540-29846-5. Springer (2005)
66. Sørmo, F., Cassens, J., Aamodt, A.: Explanation in case-based reasoning-perspectives and goals. *Artificial Intelligence Review* **24**(2), 109–143 (2005)
67. Spiekermann, S., Paraschiv, C.: Motivating Human-Agent Interaction: Transferring Insights from Behavioral Marketing to Agent Design, pp. 255–285. Kluwer Academic Publishers (2002)
68. Viappiani, P., Faltings, B., Pu, P.: Preference-based search using example-critiquing with suggestions. *Journal of Artificial Intelligence Research* **27**, 2006 (2006)
69. Weld, D., Anderson, C., Domingos, P., Etzioni, O., Lau, T., Gajos, K., Wolfman, S.: Automatically personalizing user interfaces. In: *Proceedings of the 18th International Joint Conference Artificial Intelligence (IJCAI-03)*, pp. 1613–1619. Morgan Kaufman (2003). Acapulco, Mexico
70. Williams, M.: What makes rabbit run? *International Journal of Man-Machine Studies* **21**, 333–352 (1984)
71. Williams, M., Tou, F.: Rabbit: An interface for database access. In: *Proceedings of the ACM'82 Conference*, pp. 83–87. ACM, New York, USA (1982)
72. Zhang, J., Jones, N., Pu, P.: A visual interface for critiquing-based recommender systems. In: *EC '08: Proceedings of the 9th ACM conference on Electronic commerce*, pp. 230–239. ACM, New York, NY, USA (2008)
73. Zhang, J., Pu, P.: A comparative study of compound critique generation in conversational recommender systems. In: V.P. Wade, H. Ashman, B. Smyth (eds.) *Adaptive Hypermedia and Adaptive Web-Based Systems, 4th International Conference, AH 2006, Dublin, Ireland, June 21-23, 2006, Proceedings, Lecture Notes in Computer Science*, vol. 4018, pp. 234–243. Springer (2006)

# 构建更值得信任和具有说服力的推荐系统： 特性对评估推荐系统的影响

Kyung-Hyan Yoo 和 Ulrike Gretzel

**摘要** 用户是否乐于接受推荐系统提供的推荐信息对于系统设计师和使用它们的营销人员非常重要。如果将寻求和给予建议界定为一种基本的社交行为，那么就找到了理解推荐系统说服力的重要途径。特别是研究关于特性(source characteristic)的影响，这些特征在实际人际关系的说服过程中非常丰富，因此可以为识别潜在影响因素提供重要的框架。本章回顾目前人际交互、人机交互和人与推荐系统交互场景中提到关于特性的文献，得出的结论是，就推荐系统而言，在其他环境中被确定为有影响力的许多社会线索，在推荐系统研究中尚未被引入和测试，另对推荐系统的研究和设计进行了讨论。

## 14.1 简介

推荐系统能为在线用户提供个性化的建议，在用户复杂的决策过程中[7]起到重要作用。然而，尽管推荐系统能提供基于复杂数据挖掘和分析技术的推荐，但那并不意味着由系统提供的建议就一定会被用户采用。一个建议是否被视为可靠的建议并在实际决策中加以考虑，不仅取决于用户对推荐系统的感知，而且依赖系统作为建议给予者的说服力。传统的关于说服力的文献表明用户更倾向于接受来自可靠信源的推荐[99]。因此，近年来，人们认为考虑推荐系统的可信度对提高推荐结果被接受的可能性[139, 36, 28, 56, 94]非常重要。问题是如何加强推荐系统的可信度。

近期关于技术的说服力研究表明，利用用户真实交往情境中的社会因素的技术，会更可靠和更具有说服力[88, 36]。这种概念强调了推荐系统作为准社会角色，以及有说服力的建议来源，这些建议的特征会影响用户的感知。传统的人际说服研究已经考察了多种影响力特性的作用。最重要的是，近来对人机交互环境下的研究发现，传统人际说服中起作用的这些特性在人-技术交互时也是相当重要的[36, 37, 91, 108]。就推荐系统而言，当用户对系统和推荐进行评价时[106, 76, 94, 24, 105]，一些现有的文献已研究了各种系统特性的影响。虽然这些发现提供了关于特性的好例子，这些特性有助于开发更加可信的推荐系统，但是目前仍有许多可能具有影响力的特性尚未得到检验。

因此，本章试图提供一个可信度相关研究的大纲来引起用户对特性的注意，这些特性可能在推荐系统可信度评估中起到一定的作用。为此，本章首先概述传统人际关系中发现

---

Kyung-Hyan Yoo, William Paterson University, Communication department, 300 Pompton Road, Wayne, NJ, USA, e-mail: toinette75@gmail.com

Ulrike Gretzel, TexasA&M University, 2261 TAMU, College Station, TX, USA, e-mail: ugretzel@tamu.edu

翻译：韶关学院-梁永霖老师，燕山大学-王孝先，郑州大学-吴宾 审核：刘良良，左其盛，谢冰莹，吉林大学-丁彬钊

的对寻求建议过程有影响的特性。随后讨论在人机交互情境中,特别是推荐系统领域研究过的特性。最后,本章根据还有待于在推荐系统的环境下检验的特性,确认特性研究还存在哪些缺口。总体而言,通过探究现有的发现以及识别重要的特性知识缺口,在未来研究需要关注的方面,本章将为推荐系统研究提供相关的见解。它也旨在为试图增强推荐系统可信度的设计师提供了实践意义。注意,本章关注的是推荐系统的整体认知层面。用于给出推荐的评分的信任问题会在第20章解答。

## 14.2 推荐系统作为社交角色

大部分现有推荐系统研究将推荐系统视为一种软件工具,而很大程度上忽视了它们与用户交互的这一社交角色。然而,近期研究更多地认为应该要把计算机应用(如推荐系统)看作一种社交参与者[108]。Nass和Moon[91]鼓励人们使用包括计算机在内的机器构建社交关系并在与技术的交互中应用社交规则。实际上以往许多的实证研究表明个人通过技术形成社交关系,且这些社交关系奠定了人机交互的基础[38, 81, 89, 92, 100, 107]。

最近推荐系统研究也支持“计算机是社交参与者”这一范式。例如,Wang和Benbasat[134]发现用户不仅能够认识到推荐系统具有人类特征继而将它们视为社交参与者,而且这种社交认知会影响人们对系统的评估。实验显示,用户在在线推荐系统的交互中感受到了一些人的特征,如善良、诚信等。善良和诚信是信任度的两个重要维度,而用户对系统的信任度极大地影响着用户对其可用性的感知,同时也影响用户采纳该推荐系统所推荐信息的意愿。Al-Natour、Benbasat和Cenfetelli的研究[3]认为,用户将网上购物助手也视为一个社交角色,并且为它赋予个性和行为特征。同样,Bonhard和Sasse[12]认为对推荐系统的社交背景的理解,是提供更加有用的、可信赖的以及可理解的推荐的关键。另外,Aksoy等的研究结果[2]发现,这一规则也适用于人与推荐系统之间的交互。该研究表明,当推荐系统以类似于用户决策过程的方式生成推荐时,用户会更愿意使用推荐系统。这些研究都支持了在推荐系统研究中需要关注社交因素。需要把推荐系统理解成传播源,在建立理论时运用到人际交流的理念。一套这样的理论是关于特性在说服的可能性和效果方面的影响。这些理论的基本假设是:可信的来源更具说服力。

## 14.3 来源可信度

信任度(credibility)不是一个来源的固有特性;相反,相关传播者的可信度的决策取决于消息接受者是如何感知该来源的[99, 77, 117]。因此,信息来源的信任度可以定义为消息接受者对沟通信任度的评价[37]。McGuire对信息来源可信度的研究综述[75]指出,一个可信的信息源更容易被接受,更有说服力。过去大量的研究证实,信息来源可信度与对消息接收者态度、行为目的以及行为的影响呈正相关[42, 46, 64, 118, 119]。

信任度包含了多个维度[13, 39, 102, 117]。虽然文献提出了很多种维度,但被广泛承认的关键元素有两个:可信度(trustworthiness)和专业性(expertise)[99, 37, 36, 109]。可信度和专业性已被广泛研究并在推荐系统的情境中考察[140]。

### 14.3.1 可信度

一个来源可信度指的是性格或个人诚实度等方面[99]。意图对确认来源信任度也起到一定的作用。那些具有说服意图的信息来源被认为比那些没有说服意图的更不值得信赖[102]。

因此,可信度通常都是通过某些词来描述的,如具有良好动机的、可信赖的、公正的[37]。Mayer、Davis 和 Schoorman[70]将善意(benevolence)和完整性(integrity)概念化为可信度的维度。Delgado-Ballester[26]认为可依赖性和目的性是重要的可信度维度。Fogg[36]明确了影响可信度感知的关键点:1)信息来源是公平和公正的;2)信息来源可能会与它们自己的利益不一致;3)信息来源感知相似性。在推荐系统环境中,Xiao 和 Benbasat[138]建议测试推荐系统的善意和诚信等特性,善意是指推荐系统关心用户和体现用户的兴趣的程度,诚信是指推荐系统坚持用户可接受的一套原则(如诚实)。

### 14.3.2 专业能力

Mayer 等[70]将专业能力描述为衡量信息源在某些领域产生影响的能力。Fogg 等[37]将它概念化,如知识丰富的、有经验的、能干的;因此,这一维度强调的是信息源的知识和技能。同样,O'Keefe[99]认为专业能力指的是胜任能力、专业性或资格。Fogg[36]提供了很多示例,指出专业知识的感知能力可以像宣告一个人是专家的一种标签,一种外在证据,或者是成就记录。Xiao 和 Benbasat[138]将推荐系统的能力描述为系统有效运行的能力、技能和专业知识。

### 14.3.3 对来源可信度的影响

一个信息来源是否被认为拥有专业知识和值得信任很大程度上依赖于它的特征。因此,信息特性对于用户的判断是很重要的线索。人们通常不会感知到这些线索在起作用,因为人对外界的认知处理通常通过神经末梢而不是神经中枢,在此不作论述[103]。

## 14.4 人际交互中信息特性的研究

Hovland 和他的同事[55]认为决定说服是否成功的主要刺激类型之一可以被概括为感知信息源的可察觉(observable)性。他们将来源可信度感知认定为对特定源线索观察的直接结果。许多研究人员已经调查了人际互动中影响判读信息来源可信度的各种传播者特征。

### 14.4.1 相似度

毫无疑问,在信息来源和用户之间感知到的相同点和差异点能够影响用户对信息来源可信度的判断[99]。总之,同质性理论[65]指出人们喜欢与自己相似的人。然而,相似性和信任维度之间的关系似乎比较复杂。

#### 14.4.1.1 专业性判断

过去的实证研究对相似性和来源专业性判断之间的关系并没有得到一致的结论。例如,Mills 和 Kimble[78]发现相似的人比相异的人更具有专业性。然而,Delia[27]察觉到信息来源和信息接收者之间的相似会使得信息接收者更少地把信息源当作专家。相反,一些研究发现相似度在判断信息源的专业性上不起任何作用[126, 6]。

#### 14.4.1.2 信任度判断

对信息来源感知的相似度不同,也会对传播者的感知信任度起到不同的作用。O'Keefe[99]提出,感知到的态度相似可以影响接收者对来源的喜爱,越高的信息来源的喜爱通常伴随着越高的传播者信任度判断。然而,Atkinson[6]发现种族相似和差异对信

息来源的感知可信度没有影响,而且 Delia[27]观察到相似度有时会削弱对信任的感知。

通过呈现相似度和判别交流者信任度之间关系的复杂本质, O'Keefe[99]指出感知的相似度对传播者可信度的判断取决于接收者感知到它是否以及如何与手头上问题存在相关性。因此,不同类型的相似性在不同的交流环境中可能产生不同的影响。

#### 14.4.2 喜好度

人们下意识地倾向于赞同那些看起来可爱的人[15]。喜好度指的是一个人可能会对另一个人产生的情感联结[122]。现有研究基本上支持这种假设,即让人喜爱的传播者比起那些让人讨厌的传播者对行为者有更大的影响[30, 40, 111]。喜好度被认定是一种说服策略和一种自我表现的方案[21]。O'Keefe[99]认为,对信息来源更高的喜爱度通常伴随着更高的对传播者可信度的判断。进一步,许多研究发现相似性增加了喜好度[18, 19, 52]。

同时也有证据表明,接收者对传播者的喜爱程度能够影响传播者信任度的判断,而不会影响对传播者专业性的判断[99, 66]。

#### 14.4.3 权威的象征

说服力相关的研究文献有证据表明,我们经常会依赖人脑认知捷径,觉得应该听从那些有权威象征(如某种头衔或语调)的人[109, 11, 51, 41, 104]。Hofling 等[51]发现像“Dr.”头衔等简单的东西可以让被试者认为来源是可信的,这种获取顺从的手段惊人地有效。同样,研究还表明,传播者的教育程度、职业、接受过的训练和经验等线索会影响信息接收者对来源可信度的感知。例如, Hewgill 和 Miller[50]研究了不同职业(教授和高中二年级学生)的传播者,同样信息对感知可信度的影响,发现那些被告知消息来自于教授的人认为消息来源以及消息本身都更加可信。

制服和剪裁考究的西装是另一个公认的,可以影响可信度的判断并导致盲目顺从的权威象征符[109, 21]。Bickman[11]发现,穿安全警卫制服的人比穿普通衣服的人更易驱使陌生人做事情。Sebastian 和 Bristow[116]揭示了穿着很正式的人比那些衣着休闲的人更易取得信任。

#### 14.4.4 演讲的风格

现有的很多研究表明演讲的风格也会影响对演讲者的可信度判断。例如,许多研究已证实,当研究者为他们的论点提供正反两方面的论据,而不仅仅只陈述他们支持那一面时,更能够增强他们的可信度[31, 121]。Cooper、Bennett 和 Suke[23]认为当演讲者以复杂、难以理解的方式表达时,人们对他们的专业性评价会更高。这也暗示了当专家所表达的内容无法被外行理解时,他们所具有的说服力最大[109]。也有调查发现,演讲得越不流利,演讲者的专业能力会被划分到较低的等级[73, 14, 32, 115],并且演讲的语速也会影响可信度判断,尽管这些影响并不如其他影响力证据显著[98, 1, 45, 68, 63]。同时,标示出论据的来源也会增强对表达者的专业程度和可信度的感知[35, 72]。

#### 14.4.5 外在吸引力

许多研究发现,具有外在吸引力的沟通者更具说服力[54, 123]。Eagly 等[29]指出,人们对于好看的外表会有积极的反应,好的外在可以归纳为讨人喜欢的特征,如天赋、善良、诚实和智慧等。外在吸引力产生的是间接的影响,尤其是通过影响接收者对传播者喜



爱程度[99]。

#### 14.4.6 幽默

过去的研究发现,在消息接收者评估信源可信度时,幽默感会产生一定的影响。然而,不同研究中,特定的影响也不同。许多研究表明幽默感对于传播者信任度判断有积极的影响,但是对专业性判断几乎不起任何作用[20, 44, 129]。幽默的积极影响在于增加用户对传播者的喜爱程度,且这种喜爱程度有助于增加信任度感知。相反,一些研究者却发现当过度地或者不合理地使用幽默时,幽默会减少用户对传播者的喜爱程度、信任度的感知以及信息源专业性的感知[17, 86, 130]。

### 14.5 人机交互中的特性

显然,计算机是一种工具或媒介,而不是社交生活中的行动者。然而,“媒介等同”理论表明人与计算机、电视和新媒体的交互从根本上来讲是社交化的和自然的,就像是真实生活中的互动[108]。因此这个理论表明技术应该被理解为社交参与者而不仅是工具或媒体。基于这种新的范式,越来越多的研究去关注技术中某些特定的社交属性是如何影响用户的感知和行为的。

当用户评价计算机和内容时,他们之间的相似性就非常重要[91, 36]。例如, Nass 和 Moon[91]认为,表现出类人特性的计算机更具有说服力。在他们的研究中,比起一个被动从属的计算机,占主导地位的参与者更加容易被一个主动性更强的计算机吸引以及激发自己的智慧。与占主导地位的计算机相比,从属参与者对从属计算机和主导型计算机的反应也相似,即使它们本质上具有相同的内容。Nass、Isbister 和 Lee 的研究[90]还揭示了人口统计学相似度的影响。他们的研究发现,当用户面对与自己相近的计算机智能体时,用户感知到这样的计算机智能体更具有吸引力、更值得信赖、更具说服力和更有智慧。

权威性表达在人与机器交互中被认定为是有影响力的因素。Nass 和 Moon[91]发现,与标记为普通级的电视机相比,被标记为专家级的电视被认为能够提供更好的内容。Fogg [36]也指出,承担权威角色的计算技术会更具有说服力。他认为展示所获奖项或第三方认可(如批准印章)的网站被视为具有更高的可信度。

许多研究[90, 93]表明计算机智能体的人口统计学特征影响了用户的感知。Nass 等[93]介绍,一些用户会对计算机带有性别或种族成见。研究还发现,人们认为配置男性声音的辅导计算机比设置女性声音的计算机更有能力,也更让人喜欢。他们发现,女声发音的计算机在指导爱和关系方面要比男声发音的计算机优秀,但是计算能力要比男声发音的计算机差。

此外,语言的使用,如奉承[38]、道歉[132]和礼貌[71]也会影响用户的感知和行为。此外,计算机智能体的外在吸引力也是重要的。Nass、Isbister 和 Lee[90]指出,用户更喜欢与更具吸引力的计算机智能体进行交流和互动。

最后,在人际交互环境中也验证了幽默特征。Morkes、Kernal 和 Nass[83]发现那些展示出幽默的计算机更惹人喜爱。然而,更多存在于人与人交互中的基于幽默的认同感与更深层次的交互,还没有复制到人机交互的情境中。

### 14.6 用户与推荐系统交互的特性

如果计算机被视为社交参与者,那么与推荐系统的交互也应该被理解为一种社交化的

交互。特别是那些根据用户的显式输入提供直接反馈的系统,显示出许多与社交交流相关的特性。

在现有的推荐系统文献中,许多过去的研究已经调查出推荐系统的特定属性是如何影响用户的系统评估的。Xiao 和 Benbasa[138]将研究中与推荐系统类型、输入、处理、输出有关联的多种属性进行了分类。详见第 8 章、第 15 和第 16 章。

伴随着对推荐系统行动主体的兴趣和实际使用的增长,许多研究已经调查了行动主体特征的影响。因此,接下来的部分,将会回顾这些之前确定为有影响的特性。

#### 14.6.1 推荐系统类型

推荐系统会有不同的展现形式,且可以通过过滤方法、决策策略或系统所能支持的决策数量对其进行分类[138]。之前许多研究已经讨论了不同类型的推荐系统的优点和缺点[5, 69, 16]。通过比较不同的过滤方法,发现综合了协同过滤和基于内容过滤的混合型推荐系统要比传统的单纯只用协同过滤技术的系统更有效[113, 114]。Burke[16]也证实了混合推荐系统能够对用户偏好提供更精确的预测。用于推荐系统中的不同决策策略中,补充性推荐系统要比非补充性推荐系统能激发更多的信任感,使人感知到它的有用性和功能健全性[138]。同时也会增加用户对其在产品上做决策的信心[33]。推荐系统支持的决策的数量方面,Xiao 和 Benbasat[138]认为基于需求的推荐系统相对于基于特征的推荐系统能帮助用户更好地明确自己的需求并精确回答偏好提取的问题,从而获得更好的决策质量,因而基于需求的系统更适合新手用户[34]。

#### 14.6.2 输入特性

推荐系统的输入特性包括许多相关因素,如偏好提取方法、生成新的或额外推荐的便捷性,以及当与推荐系统偏好提取界面交互时用户的控制程度等[138]。许多之前的研究发现,与推荐系统输入设计相关的特性会对系统用户的评估产生影响。Xiao 和 Benbasat[138]的研究表明,偏好提取方法(隐式与显式)影响了用户对系统的评估。隐性偏好提取方法更能让用户体现到推荐系统的易用性和满足感,相对地,显式方式被用户认为是更直接且能带更佳的决策。

给予用户更多的控制也是评价系统的一个重要影响因素。West 等[137]认为给予系统用户更多的控制将会增加他们对系统的信任感和满足感。新的研究[76]表明,比起那些系统控制以及联合控制的推荐系统,使用用户控制界面的用户对系统有更高的满足感。此外,使用用户控制界面的用户感觉到,推荐系统准确地表达了他们的品味,对系统有更高的忠诚度。Pereira[101]发现,当用户在与系统交互中控制力增强时,他们将对推荐系统表现出更积极的反应。Komiak 等[62]的研究表明,让用户控制整个过程是增强用户对虚拟行动者信任度的首要因素。Wang[133]的研究也支持了用户控制的重要性,他指出,约束较多的推荐系统对用户具有更少的信任和帮助。

除了控制,偏好提取过程的结构特征(相关性、透明性和使用成本)也会影响用户对推荐系统的感知[43]。Gretzel 和 Fesenmaier 的研究发现提取过程的主题相关性、透明性和用户输入的操作成本显著地影响了用户对提取过程的认知和评价。这些发现表明,通过回答问题,系统承担了一种社交角色,表达了交流用户兴趣偏好,这都被认为是有价值的。回答的问题越多,就越能提供有价值的反馈。同时,在交互过程中显式的引导也很重

要。虽然没有专门测量过信任程度,但善意和直接意图是信任重要的驱动因素,并且也暗含在过程透明性之中。McGinty 和 Smyth[74]表明,推荐系统的对话风格在输入过程中也是很重要的。相反[43],他们认为基于对比的推荐方法(要求用户从一系列的推荐项目中选择他们偏爱的项目,而非深度对话方法直接询问用户关于产品特色重要性的问题)能最小化用户的操作成本,且同时保证推荐质量。

### 14.6.3 过程特性

推荐系统在计算推荐过程中显示出的特性似乎影响着用户对系统的认知[138]。该过程因素包含关于搜索过程和系统响应时间的信息。Mohr 和 Bitner[79]指出,系统用户使用各种线索或指标来评估通过决策辅助手段节省的工作量。告知用户关于搜索过程的指标有助于用户意识到系统为用户节省的工作量。用户对通过决策辅助手段节省的工作量有着越多认知,那么他们对决策过程就会越满意[9]。Sutcliffe 等[125]发现了用户对未提供搜索过程指示的信息检索系统提出了可用性或理解的问题。

有研究考虑了系统响应时间的影响,即用户输入和系统响应的间隔时间是非常重要的。Basartan[8]通过改变一个模拟的导购网站的响应时间,发现用户更不喜欢那些要等很长时间才能获得推荐结果的网站。相反,Swearingen 和 Sinha[128]发现用户用在注册和接收来自系统推荐的时间在系统用户认知方面却没有明显的影响。较长的注册过程提高用户对系统的满意度和认可度[76]。Xiao 和 Benbasat[138]解释,先前关于系统响应时间矛盾的研究成果可能取决于用户的成本收益评估。当用户把等待的效益看成高质量的推荐时,他们认为用户不会对推荐系统形成负面的评价。一些关于提取成本以及提取过程中的认知价值之间的关系研究结果[43]都支持该假设。

### 14.6.4 输出特性

推荐系统输出阶段描述的特性与呈递给用户的推荐内容以及展示格式相关联。现有研究表明推荐的内容和格式对用户评估推荐系统具有重大的影响[135, 138, 120, 24]。Xiao 和 Benbasat[138]指出,当用户对推荐系统进行评价时,推荐内容的三大重要方面:推荐选择的熟悉度、推荐产品的信息数量以及对于推荐结果的解释,是非常相关的。

研究发现更熟悉的推荐结果能提高用户对推荐系统的信任度。Swearingen 和 Sinha[120]发现为用户推荐熟悉的商品有助于建立用户对推荐系统的信任。Cooke 等[22]的一项研究也表明推荐不熟悉的物品会降低用户对推荐系统的评价。进一步,获得产品的信息能够用户对推荐系统的评价产生积极的影响。Sinha 和 Swearingen[120]表明,推荐网页上有详细的产品信息能加强用户对推荐系统的信任度。Cooke 等[22]也说明,如果推荐系统提供了新产品的详细信息,将会提高推荐给用户的不熟悉产品的吸引力。

目前已有大量研究探讨有关推荐系统用户评价解释的影响,Wang 和 Benbasat[135]发现对推荐系统合理逻辑的解释能够增强用户对推荐系统能力和行为的信心。Herlocker 等[48]也认为解释推荐结果对于建立对系统的信任非常重要,因为当用户不明白为什么某些物品被推荐给他们时,他们是不太可能相信这些推荐的。为了提高用户对系统的信任度,推荐系统必须通过解释内容来建立用户和系统之间的连接。同样,其他研究[105, 131]也表明在有解释界面的情况下用户表现出更多的信任感。

展示给用户的推荐结果的格式也会影响推荐系统的用户评价。Sinha 和 Swearingen[120]

表明推荐结果展示界面的导航和布局明显影响了用户对系统的满意度。Sinha 和 Swearingen [128]进一步研究发现,界面导航和布局设计也影响了用户对系统的综合评价。无独有偶,Yoon 和 Lee[142]也表明界面设计和展示格式影响了系统用户的行为。然而,Bharti 和 Chaudhury[10]的一项研究却发现关于用户的满意度,导航效果不起任何明显作用。

此外,Schafer[112]建议在一个界面内,合并偏好界面和推荐引导界面可以使推荐系统看起来更有帮助,因为这种新的“动态查询”界面能提供关于个人偏好改变的影响的及时反馈。因为它合并了输入和输出的界面,该建议涉及的提示(如透明度)已经在输入特性中讨论过。

#### 14.6.5 内嵌的智能体特性

推荐系统通常包含虚拟角色来指导用户完成操作过程。可以假设如果推荐系统更加人性化,那么社交反应将会更普遍。事实上,大量的研究强调在推荐系统的上下文中,内嵌的智能体起到重要的作用。例如,已发现系统接口中人性化虚拟智能体的存在既提高系统可信度[85]、增加社交化[106]、增强网上购物体验[53],又增强信任[136]。随着对这种接口智能体的感兴趣,很多研究已经开始研究接口智能体的某些特性如何影响推荐系统用户的认知和评价。

智能体的重要识别特性之一是拟人化。拟人化的定义是拥有人类外表和行为属性的程度[96, 97, 60, 95]。许多研究者发现内嵌智能体的拟人化可影响用户与计算机的交互(如[96, 60, 95]),特别是与推荐系统的交互[106]。至今,拟人化的智能体的优势和开销一直备受争议。例如,在一些研究中发现,相比于那些不太拟人化的接口智能体,更拟人化的接口智能体,更可信、更迷人、更吸引人、更令人喜爱[97, 60],然而其他一些研究结果却得到相反的结果[96, 95, 87]。与这种智能体的社交行为可能会让用户增加对实际系统功能的期望。

人的声音是一个非常强大的社交因素,已经深入影响了人类的交互[88]。然而,在嵌入的接口智能体的上下文中研究结果并不普及并且目前尚未定论。在一些研究中,发现接口智能体的声音输出特性有助于诱导社交和引起用户的情感表达[106, 82]。但是其他研究却发现当系统智能体只用文字沟通时的社交化水平更高[124]。同时发现接口智能体的人口统计特征也影响了系统用户的感知和行为。Qiu[106]表示,当智能体符合他们的种族特性、性别等时,系统用户评价系统更善于交际、更有能力和更令人享受从而支持同质性假说。Cowell 和 Stanny[25]也观察到系统用户更喜欢与他们种族匹配的并且看上去更有活力的对象交互。Nowak 和 Rauh[97]的研究表明人们明显倾向于选择匹配他们性别的特征的推荐。

除了以上类似因素之外,在嵌入的接口智能体的研究中也调查了其他特性。Holzwarth 等[53]已测试了智能体吸引性和专业性的作用。他们发现,在产品加入中等水平智能体可以作为一个更有效的销售智能体,而在高水平产品涉入中专家智能体则是一个更有效的说服者。而且,Cowell 和 Stanney[25]强调了接口智能体中非语言行为像面部表情、眼神交会、手势、副语言交际和姿势等的潜在影响。然而,目前对这一领域的研究还是非常有限的。

#### 14.7 讨论

Swearingen 和 Sinha[128]表示,推荐系统的最终效果将依赖于超出算法质量的诸多因素。然而,推荐系统特性通常得以实现是因为它们可以被实施执行。这些特性可能会在系统总体评价或可用性研究中被测试,但是几乎很少会根据其说服力进行评估。Haubl 和



Murray[47]证明在实时推荐中,推荐系统的确对消费者偏好和选择产生深刻影响。因此,将推荐系统界定为不但是社交角色,而且是说服力角色,这对理解他们潜在因素至关重要。上述文献中表明大量的推荐系统特性可能都具有影响力。遵循“计算机作为社交行动者”的模式[108, 36],最近的推荐系统研究已经开始强调推荐系统社交方面的作用,并强调整合社交特征以创造更可信的和更有说服力的系统[134, 106, 3]。对推荐系统“作为社交参与者”的这种认识对推荐系统的研究和设计有重大意义。最重要的是,将人与推荐系统之间的交互界定为一种社交交换意味着,那些在传统的寻求建议关系中被确定为有影响力的重要特性,在拟人化推荐系统交互中也被视为有潜在影响力。

## 14.8 影响

在评价推荐系统时,理解相关特性的影响有许多理论价值和实践意义。从理论角度看,经典的人际沟通理论需要扩大范围和应用来理解人与推荐系统的关系。通过应用经典理论,研究人员可以测试并分析人与推荐系统交互的多个方面。然而,当应用这些理论和开发方法论来测试它们时,应该考虑人与推荐系统交互的独有特质。进一步,虽然推荐系统相关的研究中存在一些有关特性的研究,然而目前这些并不系统,甚至有些是有争议的。显然,在这一领域还需做更多的研究以便构建一个强大的理论框架。

从实际价值角度来看,理解推荐系统作为拥有影响用户感知特征的社交参与者这一角色,有助于系统开发人员和设计人员更好地理解用户与系统之间的交互。社交交互性靠用户信任度茁壮成长,同时也受到说服力的影响。在“引导偏好的方式”和“得出推荐的方式”这些过程中,用户越富有洞察力,那么一个推荐系统将被感知到更高的可信度,同时也越容易被接受[43]。混合系统、显式的启发并且能够让用户控制整个过程似乎是非常有效的策略[113, 114, 138, 16, 137, 101]。Schafer[112]提出的动态查询接口合并了偏好接口和推荐引导接口,可能是一种有助于用户感觉到自己在操控系统的方式,因为这种动态查询接口能够根据用户偏好的改变即时响应。在推荐系统交互中,响应时间需保持简短[8]和搜索过程细节需告知用户以便展示系统的工作[79, 9, 125],因为这将影响可信度感知。当生成推荐结果时,更常见的详细产品描述推荐[22]和关于推荐是如何产生的相关底层逻辑的解释[133, 48]将增加用户对系统的感知可信度。使用由复杂的数据挖掘技术所挖掘的历史记录和模式更好地理解用户系统,将帮助系统产生更多熟悉的推荐给用户。连同推荐产品的文本描述,推荐系统设计者可以考虑提供虚拟的产品体验。Jiang 和 Benhasat[56]指出虚拟产品的体验提高消费者的产品认知度、品牌态度、购买意愿和降低感知的风险。添加虚拟产品体验不仅能够使用户更好地了解推荐产品,而且能够激起他们对产品更大的关注、兴趣和乐趣。推荐系统设计人员还应注意推荐的显示格式[128, 142]。当推荐呈现给用户时,导航效果和设计熟悉度以及吸引力度也应被考虑。

最重要的是,研究推荐系统环境中相关的特性为可信的和有说服力的推荐系统相关设计提供了启发。设计面临的挑战是寻找如相似度、亲和力和权威性 etc 等能够被操作并转换成具体的设计特点的方法,从而适用于推荐系统的环境。例如,呈现系统的信任权威的第三方印章能提高系统整体的信誉度。通过引导用户的产品偏好和选择用户倾向的决策策略,以基于需要问题的使用,实现推荐系统和用户之间的相似性[138]。通过不同的沟通风格和声音特征来操纵推荐系统的个性(像外向或内向)与用户个性相匹配[49, 80]。为系统接口添加一个嵌入的智能体,通过这种方式,某些特征更容易被实现。嵌入的智能体作为系统的代理人,因此,强调系统建议提供者这一社交角色[141]。语音接口可以成为将特性



转换为取得信任的推荐系统设计的另一种方式。

从市场营销的角度来看,创造出具有可靠性和诱导性的推荐系统是关键,因为它扮演着与实体店销售员类似的角色,即需要与顾客互动以及根据顾客的购买情况提供建议[134, 62]。因此,创造出更具社交化和更值得信赖的推荐系统将有助于市场营销人员加强他们的电子化服务。

## 14.9 未来研究方向

尽管现有研究已经在人与推荐系统的寻求建议关系中发现并检测了很多有影响力的特性,但仍有通用传播理论提出的许多潜在特性没有得到验证,比如,权威性、关怀、像面部表情和姿势这样的非语言行为,以及幽默。在未来推荐系统研究中,这些未被测试的特性需要被成功地实现并进行实证检验。

已经发现和测试的特性还需要被精确地检测。对于判断可信度的作用往往是较复杂的,而不是与之前讨论的人与人寻求建议的过程中发现的影响呈线性关系[99]。因为情境因素、个人差异以及产品类型等关系在决定推荐系统的可信度性时也能扮演重要的角色,所以从设计方面考虑,对于一个能够提供精确输入的特定推荐系统来说,必须专门对这些关系进行测试。

此外,可能会存在另外的特性,它们对寻求人际关系的影响可能不是很突出,但在推荐系统领域里,却是要被考虑的重要方面。例如,技术人性化已被认识到是在技术交互中具有重要影响的特性[60, 96],但却不是影响人类之间交互的重要特性。界面智能体的真实性也被认为是一个潜在的有影响力的特性。有证据表明:本来用户会更倾向于和拟人化的软件角色交互,但如果这个角色设计得不好,用户进行互动的意愿也不会强,反而不如一个设计得很好的非拟人的软件角色,如狗[59]。在未来的研究中,这种额外的源信号需要被识别和测试。

目前,一些特性已被单独地测试过。但为了研究它们相互作用的影响,可能应该同时对不同源信号进行测试。这将有助于理解各种特性之间的联系。

总而言之,本章提到的文献表明:对这一领域的研究是迫在眉睫的。它还表明,新方法可能需要进一步完善以研究在潜意识层面的影响。特别地,如果要评估具有说服力的推荐系统,那么更加注重对推荐结果接受行为的评估似乎是有道理的。

## 参考文献

1. Addington, D.W. (1971). The effect of vocal variations on ratings of source credibility. *Speech Monographs*, 38, 242-247.
2. Aksoy, L., Bloom, P. N., Lurie, N. H., & Cooil, B. (2006). recommendation agents think like people? *Journal of Service Research*, 8(4), 297-315.
3. Al-Natour, S., Benbasat, I., & Cenfetelli, R. T. (2006). The role of design characteristics in shaping perceptions of similarity: The case of online shopping assistants. *Journal of Association for Information Systems*, 7 (12), 821-861.
4. Andersen, K.E., & Clevenger, T., Jr. (1963). A summary of experimental research in ethos. *Speech Monographs*, 30, 59-78.
5. Ansari, A., Essegaier, S. and Kohli, R. (2000). Internet Recommendation Systems. *Journal of Marketing Research*, 37(3), 363-375
6. Atkinson, D.R., Winzelberg, A., & Holland, A. (1985). Ethnicity, locus of control for family planning, and pregnancy counselor credibility. *Journal of Counseling Psychology*, 32, 417-421.
7. Barwise, P., Elberse, A. and Hammond, K. (2002). Marketing and the internet: a research

- review. In B. Weitz, & R. Wensley (Eds), *Handbook of Marketing* (pp. 3-7), New York, NY, Russell Sage.
8. Basartan, Y. (2001). *Amazon Versus the Shopbot: An Experiment About How to Improve the Shopbots*. Ph.D. Summer Paper, Carnegie Mellon University, Pittsburgh, PA.
  9. Bechwati, N. N., & Xia, L. (2003). Do computers sweat? The impact of perceived effort of on-line decision aids on consumers' satisfaction with the decision process. *Journal of Consumer Psychology*, 13, 1-2, 139-148.
  10. Bharti, P., & Chaudhury, A. (2004). An Empirical Investigation of Decision-Making Satisfaction in Web-Based Decision Support Systems. *Decision Support Systems*, 37 (2), 187-197.
  11. Bickman, L. (1974). The social power of a uniform. *Journal of Applied Social Psychology*, 4, 47-61.
  12. Bonhard P., & Sasse M. A. (2005). I thought it was terrible and everyone else loved it - A New Perspective for Effective Recommender System Design, in *Proceedings of the 19th British HCI Group Annual Conference* (pp. 251-261), Napier University, Edinburgh, UK 5-9 September 2005.
  13. Buller, D. B., & Burgoon, J. K. (1996). Interpersonal deception theory. *Communication Theory*, 6, 203-242.
  14. Burgoon, J. K., Birk, T., & Pfau, M. (1990). Nonverbal behaviors, persuasion, and credibility. *Human Communication Research*, 17, 140-169.
  15. Burgoon, J. K., Dunbar, N. E. & Segring, C. (2002). Nonverbal Influence. In J. P. Dillard, & M. Pfau (Eds), *Persuasion Handbook: Developments in Theory and Practice* (pp.445-473). Thousand Oaks, CA: Sage Publications.
  16. Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.
  17. Bryant, J., Brown, D., Silberberg, A.R., & Elliott, S.M. (1981). Effects of humorous illustrations in college textbooks. *Human Communication Research*, 8, 43-57.
  18. Byrne, D. (1971). *The attraction paradigm*. New York: Academic Press.
  19. Carli, L. L., Ganley, R., & Pierce-Otay, A. (1991). Similarity and satisfaction in roommate relationships. *Personality and Social Psychology Bulletin*, 17 (4), 419-426.
  20. Chang, K.-J., & Gruner, C. R. (1981). Audience reaction to self-disparaging humor. *Southern Speech Communication Journal*, 46, 419-426.
  21. Cialdini, R. B. (1994). Interpersonal Influence. In S. Shavitt, & T. C. Brock (Eds). *Persuasion: Psychological Insights and Perspective* (pp.195-217). Needham Heights, Massachusetts, Allyn and Bacon
  22. Cooke, A. D. J., Sujana, H., Sujana, M., & Weitz, B. A. (2002). Marketing the Unfamiliar: The Role of Context and Item-Specific Information in Electronic Agent Recommendations, *Journal of Marketing Research*, 39 (4), 488-497.
  23. Cooper, J., Bennett, E.A., & Sukel, H. L. (1996). Complex scientific testimony: How do jurors make decisions? *Law and Human Behavior*, 20, 379-394.
  24. Cosley, D., Lam, S. K., Albert, I., Konstan, J., & Riedl, J. (2003). Is seeing believing? How recommender systems influence users' opinions. In *Proceedings of CHI 2003: Human Factors in Computing Systems* (pp. 585-592). New York: ACM Press.
  25. Cowell, A. J. & Stanney, K. M. (2005). Manipulation of non-verbal interaction style and demographic embodiment to increase anthropomorphic computer character credibility. *International Journal of Human-Computer Studies*, 62, 281-306.
  26. Delgado-Ballester, E. (2004). Applicability of a brand trust scale across product categories: A multigroup invariance analysis. *European Journal of Marketing*, 38(5/6), 573-592.
  27. Delia, J. G. (1975). Regional dialect, message acceptance, and perceptions of the speaker. *Central States Speech Journal*, 26, 188-194.
  28. Dijkstra, J. J., Liebrand, W. B. G., & Timminga, E. (1998). Persuasiveness of Expert Systems. *Behaviour & Information Technology*, 17(3), 155-163.
  29. Eagly, A. H., Ashmore, R. D., Makhijani, M. G., & Longo, L. C. (1991). What is beautiful is good, but: A meta-analytic review of research on the physical attractiveness stereotype. *Psychological Bulletin*, 110, 109-128.
  30. Eagly, A. H. & Chaiken, S. (1975). An attribution analysis of the effect of communicator characteristics on opinion change: The case of communicator attractiveness, *Journal of Personality and Social Psychology*, 32(1), 136-144.
  31. Eagly, A.H., Wood, W., & Chaiken, S. (1978). Causal inferences about communicators and their effect on opinion change. *Journal of Personality and Social Psychology*, 36, 424, 435.
  32. Engstrom, E. (1994). Effects of nonfluencies on speakers' credibility in newscast settings. *Perceptual and Motor Skills*, 78, 739-743.

33. Fasolo, B., McClelland, G. H., & Lange, K. A. (2005). The Effect of Site Design and Inter-attribute Correlations on Interactive Web-Based Decisions. In C. P. Haughvedt, K. Machleit, and R. Yalch (Eds.) *Online Consumer Psychology: Understanding and Influencing Behavior in the Virtual World* (pp. 325-344). Mahwah, NJ: Lawrence Erlbaum Associates.
34. Felix, D., Niederberger, C., Steiger, P., and Stolze, M. (2001). Featur-Oriented vs. Needs-Oriented Product Access for Non-expert Online Shoppers. In B. Schmid, K. Stanoevska-Slabeva, and V. Tschammer-Zurich (Eds.) *Towards the E-Society: E-Commerce, E-Business, and E-Government* (pp. 399-406). New York: Springer.
35. Fleshler, H., Ilardo, J., & Demoretcky, J. (1974). The influence of field dependence, speaker credibility set, and message documentation on evaluations of speaker and message credibility. *Southern Speech Communication Journal*, 39, 389-402.
36. Fogg, B. J. (2003). *Persuasive technology: Using computers to change what we think and do*. San Francisco: Morgan Kaufmann.
37. Fogg, B. J., Lee, E., & Marshall, J. (2002). Interactive technology and Persuasion. In J. P. Dillard, & M. Pfau (Eds). *Persuasion handbook: Developments in theory and practice* (pp.765-797). London, United Kingdom: Sage
38. Fogg, B.J., & Nass, C. (1997). Silicon sycophants: Effects of computers that flatter, *International Journal of Human-Computer Studies*, 46(5), 551-561.
39. Gatignon, H., & Robertson, T. S. (1991). *Innovative decision processes*. Englewood Cliffs, NJ: Prentice Hall.
40. Giffen, K., & Ehrlich, L. (1963). Attitudinal effects of a group discussion on a proposed change in company policy. *Speech Monographs*, 30, 377-379.
41. Giles, H., & Coupland, N. (1991). *Language: Contexts and consequences*. Pacific Grove, CA: Brooks/Cole.
42. Gilly, M. C., Graham, J. L., Wolfenbarger, M. F., & Yale, L. J. (1998). A dyadic study of personal information search. *Journal of the Academy of Marketing Science*, 26(2), 83-100.
43. Gretzel, U., & Fesenmaier, D. R. (2007). Persuasion in Recommender Systems. *International Journal of Electronic Commerce*, 11(2), 81-100.
44. Gruner, C. R., & Lampton, W. E. (1972). Effects of including humorous material in a persuasive sermon. *Southern Speech Communication Journal*, 38, 188-196.
45. Gundersen, D.F., & Hopper, R. (1976). Relationships between speech delivery and speech effectiveness. *Communication Monographs*, 43, 158-165.
46. Harmon, R. R., & Coney, K. A. (1982). The persuasive effects of source credibility in buy and lease situations. *Journal of Marketing Research*, 19(2), 255-260.
47. Häubl and Murray (2003). Preference Construction and Persistence in Digital Marketplaces: The Role of Electronic Recommendation Agents. *Journal of Consumer Psychology*, 13 (1&2), 75-91.
48. Herlocker, J., Konstan, J. A., & Riedl, J. (2000). Explaining Collaborative Filtering Recommendations. *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, Philadelphia, PA, 241-250.
49. Hess, T. J., Fuller, M. A. & Mathew, J. (2005). Involvement and Decision-Making Performance with a Decision Aid: The Influence of Social Multimedia, Gender, and Playfulness. *Journal of Management Information Systems*, 22(3), 15-54.
50. Hewgill, M. A., & Miller, G. R. (1965). Source credibility and response to fear-arousing communications. *Speech Monographs*, 32, 95-101.
51. Hofling, C. K., Brotzman, E., Dalrymple, S., Graves, N., & Pierce, C. M. (1966). An experimental study of nurse-physician relationships. *Journal of Nervous and Mental Disease*, 143, 171-180.
52. Hogg, M. A., CooperShaw, L., & Holzworth, D. W. (1993). Group prototypically and depersonalized attraction in small interactive groups. *Personality and Social Psychology Bulletin*, 19 (4), 452-465.
53. Holzwarth, M., Janiszewski, C. & Neumann, M.M. (2006). The influence of avatars on online consumer shopping behavior. *Journal of Marketing*, 70(October), 19-36.
54. Horai, J., Naccari, N., & Fatoullah, E. (1974). The effects of expertise and physical attractiveness upon opinion agreement and liking. *Sociometry*, 37, 601-606.
55. Hovland, C.I., Janis, I.L., & Kelley, H.H. (1953). *Communication and Persuasion*. New Haven, CT: Yale University Press.
56. Jiang, J. J., Klein, G., & Vedder, R.G. (2000). Persuasive expert systems: The influence of confidence and discrepancy. *Computers in Human Behavior*, 16, 99-109.
57. Jiang, Z. & Benbasat, I. (2005). Virtual Product Experience: Effects of Visual and Functional Control of Products on Perceived Diagnosticity and Flow in Electronic Shopping. *Journal of*

- Management Information Systems*, 21 (3), 111-148.
58. Kim, B.-D., & Kim, S.-O. (2001). A new recommender system to combine content-based and collaborative filtering systems. *Journal of Database Marketing*, 8(3), 244-252.
  59. Kiesler, S., Sproull, L., & Waters, K. (1996). A prisoner's dilemma experiment on cooperation with people and human-like computers. *Journal of Personality and Social Psychology*, 70(1), 47-65.
  60. Koda, T. (1996). *Agents with faces: A study on the effects of personification of software agents*. Master's Thesis, Massachusetts Institute of Technology, Boston, MA, USA.
  61. Komiak, S. X. & Benbasat, I. (2004). Understanding customer trust in agent-mediated electronic commerce, web-mediated electronic commerce and traditional commerce. *Information Technology and Management*, 5 (1&2), 181-207
  62. Komiak, S. Y. X., Wang, W., & Benbasat, I. (2005). Trust Building in Virtual Salespersons Versus in Human Salespersons: Similarities and Differences. *e-Service Journal*, 3(3), 49-63.
  63. Lautman, M. R., & Dean, K. J. (1983). Time compression of television advertising. In I. Percy & A. G. Woodside (Eds.), *Advertising and consumer psychology* (pp.219-236). Lexington, Ma: Lexington Books.
  64. Lascu, D.-N., Bearden, W. O., & Rose, R. L. (1995). Norm extremity and personal influence on consumer conformity. *Journal of Business Research*, 32(3), 201-213.
  65. Lazarsfeld, P., and R. K. Merton. (1954). Friendship as a Social Process: A Substantive and Methodological Analysis. In M. Berger, T. Abel, and C. H. Page (Eds), *Freedom and Control in Modern Society* (pp. 18-66). New York: Van Nostrand.
  66. Levine, R. V. (2003). Whom Do We Trust? Experts, Honesty, and Likability. In R. V. Levine (Ed.), *The Power of Persuasion*, pp.29-63. Hoboken, NJ: John Wiley & Sons.
  67. Liao, Q. (2005). *Empirical findings on persuasiveness of recommender systems for customer decision support in electronic commerce*. Doctoral dissertation, Mississippi State University.
  68. MacLachlan, J. (1982). Listener perception of time-compressed spokespersons. *Journal of Advertising Research*, 22(2), 47-51.
  69. Maes, P., Guttman, R. H., & Moukas, A.G. (1999). *Agents that Buy and Sell*, Communication of the ACM, 42(3), 81-91.
  70. Mayer, R. C., Davis, J. H. & Schoorman, F. D. (1995). An integrative model of organizational trust. *Academy of Management Review*, 20, 709-734.
  71. Mayer, R. E., Johnson, W. L., Shaw, E. & Sandhu, S. (2006). Constructing computer-based tutors that are socially sensitive: Politeness in educational software. *International Journal of Human-Computer Studies*, 64(1), 36-42.
  72. McCroskey, J. C. (1970). The effects of evidence as an inhibitor of counter-persuasion. *Speech Monographs*, 37, 188-194.
  73. McCroskey, J. C., & Mehrley, R. S. (1969). The effects of disorganization and nonfluency on attitude change and source credibility. *Speech Monographs*, 36, 13-21.
  74. McGinty, L. & Smyth, B. (2002). Deep Dialogue vs Casual Conversation in Recommender Systems. In F. Ricci and B. Smyth (Eds), *Proceedings of the Workshop on Personalization in eCommerce at the Second International Conference on Adaptive Hypermedia and Web-Based Systems* (AH 2002), pp. 80-89. Universidad de Malaga, Malaga, Spain, Springer.
  75. McGuire, W. J. (1968). The Nature of Attitudes and Attitude Change. In G. Lindzey and E. Aronson (Eds.), *Handbook of Social Psychology*. Reading, MA: Addison-Wesley.
  76. McNee, S. M.; Lam, S. K.; Konstan, J. A.; and Riedl, J. (2003). *Interfaces for eliciting new user preferences in recommender systems*. In User Modeling 2003, LNCS 2702,,178-187. Springer.
  77. Michener, H. A., DeLamater, J. D., & Myers, D. J. (2004). *Social Psychology*. Wadsworth: Thomson Learning, Inc.
  78. Mills, J., & Kimble, C. E. (1973). Opinion change as a function of perceived similarity of the communicator and subjectivity of the issue. *Bulletin of the psychonomic society*, 2, 35-36.
  79. Mohr, L. A., & Bitner, M. J. (1995). The Role of Employee Effort in Satisfaction with Service Transactions. *Journal of Business Research*, 32(3), 239-252.
  80. Moon, Y. (2002). Personalization and Personality: Some Effects of Customizing Message Style Based on Consumer Personality. *Journal of Consumer Psychology*, 12(4), 313-326.
  81. Moon, Y., & Nass, C. (1996). How "real" are computer personalities? Psychological responses to personality types in human-computer interaction. *Communication Research*, 23(6), 651-674.
  82. Moreno, R., Mayer, R. E., Spire, H. A., & Lester, J. C. (2001). The case for social agency in computer-based teaching: Do students learn more deeply when they interact with animated pedagogical agents? *Cognition and Instruction*, 19(2), 177-213.

83. Morkes, J., Kernal, H. K., & Nass, C. (1999). Effects of humor in task-oriented human-computer interaction and computer-mediated communication: A direct test of SRCT theory. *Human-Computer Interaction*, 14(4), 395-435.
84. Moulin, B., Irandoust, H., Belanger, M., & Desbordes, G. (2002). Explanation and argumentation capabilities: Towards the creation of more persuasive agents. *Artificial Intelligence Review*, 17, 169-222.
85. Moundridou, M., & Virvou, M. (2002). Evaluation the persona effect of an interface agent in a tutoring system. *Journal of Computer Assisted Learning*, 18(3), 253-261.
86. Munn, W. C., & Gruner, C. R. (1981). "Sick" Jokes, speaker sex, and informative speech. *Southern Speech Communication Journal*, 46, 411-418.
87. Murano, P. (2003). Anthropomorphic vs. Non-Anthropomorphic Software Interface Feedback for Online Factual Delivery. *Proceedings of the Seventh International Conference on Information Visualization*. Retrieved October, 1, 2008 from <http://portal.acm.org/citation.cfm?id=939634&dl=ACM&coll=portal>
88. Nass, C., & Brave, S. (2005). *Wired for Speech: How Voice Activates and Advances the Human-Computer Relationship*. Cambridge, MA: MIT Press.
89. Nass, C., Fogg, B.J., & Moon, Y. (1996). Can computers be teammates? *International Journal of Human-Computer Studies*, 45(6), 669-678.
90. Nass, C., Isbister, K., & Lee, E. -J. (2000). Truth is beauty: Researching embodied conversational agents. In J. Cassell, J. Sullivan, S. Prevost, & E. Churchill (Eds), *Embodied conversational agents* (pp. 374-402). Cambridge, MA: MIT Press
91. Nass, C. & Moon, Y. (2000). Machines and Mindlessness: Social Responses to Computers. *Journal of Social Issues*, 56 (1), 81-103.
92. Nass, C., & Moon, Y., & Carney, P. (1999). Are respondents polite to computers? Social desirability and direct responses to computers. *Journal of Applied Social Psychology*, 29 (5), 1093-1110.
93. Nass, C., Moon, Y., & Green, N. (1997). Are computers gender-neutral? Gender stereotypic responses to computers. *Journal of Applied Social Psychology*, 27(10), 864-876.
94. Nguyen, H., Masthoff, J. & Edwards, P. (2007). Persuasive Effects of Embodied Conversational Agent Teams. *Proceedings of 12th International Conference on Human-Computer Interaction*, Beijing, China, Springer-Verlag, Berlin, 176-185
95. Nowak, K. (2004). The influence of anthropomorphism and agency on social judgment in virtual environments. *Journal of Computer-Mediated Communication*, 9 (2)
96. Nowak, K. L., & Biocca, F. (2003). The effect of the agency and anthropomorphism on user's sense of telepresence, copresence, and social presence in virtual environments. *Presence:Teleoperators and Virtual Environments*, 12(5), 481-494.
97. Nowak, K., & Rauh, C. (2005). The influence of the avatar on online perceptions of anthropomorphism, androgyny, credibility, homophily, and attraction. *Journal of Computer-Mediated Communication*, 11 (1)
98. O'Keefe, D. J. (1998). Justification explicitness and persuasive effect: A meta-analytic review of the effects of varying support articulation in persuasive messages. *Argumentation and advocacy*, 35, 61-75.
99. O'Keefe, D. J. (2002). *Persuasion: Theory & Research*. Thousand Oaks, CA: Sage Publications.
100. Parise S, Kiesler S, Sproull L, Waters K. (1999). Cooperating with life-like interface agents. *Computers in Human Behavior*, 15, 123-142.
101. Pereira, R. E. (2000). Optimizing Human-Computer Interaction for the Electronic Commerce Environment. *Journal of Electronic Commerce Research*, 1 (1), 2000, 23-44.
102. Petty, R. E., & Cacioppo, J. T. (1981). *Attitudes and persuasion: Classic and contemporary approaches*. Dubuque, IA: William C. Brown.
103. Petty, R. E., & Cacioppo, J. T. (1986). *Communication and Persuasion: Central and Peripheral Routes to Attitude Change*. New York: Springer-Verlag.
104. Pittam, J. (1994). *Voice in social interaction: An interdisciplinary approach*. Thousand Oaks, CA: Sage.
105. Pu, P. & Chen, L. (2007). Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems*, 20, 542-556
106. Qiu, L. (2006). *Designing social interaction with animated avatars and speech output for product recommendation agents in electronic commerce*. Doctoral Thesis, University of British Columbia, Vancouver.
107. Quintanar, L. R., Crowell, C. R., Pryor, J. B., & Adamopoulos, J. (1982). Human-computer interaction: A preliminary social psychological analysis. *Behavior Research Methods & In-*



- strumentation, 14 (2), 210-220.
108. Reeves, B. & Nass, C. (1996). *The media equation: how people treat computers, television, and new media like real people and places*. New York, NY: CSLI.
109. Rhoads, K. V., & Cialdini, R. B. (2002). The Business of Influence. J. P. Dillard & M. Pfau (Eds). *Persuasion handbook: Developments in theory and practice*. (pp.513-542). London, United Kingdom: Sage
110. Rokach, L. (2008), Mining manufacturing data using genetic algorithm-based feature set decomposition, *Int. J. Intelligent Systems Technologies and Applications*, 4(1):57-78.
111. Sampson, E. E., & Insko, C. A. (1964). Cognitive consistency and performance in the autokinetic situation. *Journal of Abnormal and Social Psychology*, 68, 184-192.
112. Schafer, J. B. (2005). DynamicLens: A Dynamic User-Interface for a Meta-Recommendation System. *Workshop: Beyond Personalization 2005, IUI'05*, San Diego, CA.
113. Schafer, J. B., Konstan, J.A., & Riedl, J. (2002). Meta-Recommendation Systems: User-Controlled Integration of Diverse Recommendations. Paper presented at the 11th international Conference on Information and Knowledge Management, McLean, VA, November 2002.
114. Schafer, J.B., Konstan, J.A., & Reidl, J. (2004). The View through MetaLens: Usage Patterns for a Meta Recommendation System. *IEE Proceedings Software*.
115. Schliesser, H. F. (1968). Information transmission and ethos of a speaker using normal and defective speech. *Central States Speech Journal*, 19, 169-174.
116. Sebastian, R. J., & Bristow, D. (2008). Formal or Informal? The Impact of Style of Dress and Forms of Address on Business Students' Perceptions of Professors. *Journal of Education for Business*, 83(4), 196-201.
117. Self, C. S. (1996). Credibility. In M. B. Salwen & D. W. Stacks (Eds.), *An integrated approach to communication theory and research* (pp. 421-441). Mahwah, NJ: Lawrence Erlbaum.
118. Sénécal, S., & Nantel, J. (2003). Online influence of relevant others: A framework. *Proceedings of the Sixth International Conference on Electronic Commerce Research (ICECR-6)*, Dallas, Texas.
119. Sénécal, S., & Nantel, J. (2004). The influence of online product recommendations on consumers' online choices. *Journal of Retailing*, 80(2), 159-169.
120. Sinha, R. & Swearingen, K. (2001). Comparing Recommendations Made by Online Systems and Friends. *Proceedings of the 2nd DELOS Network of Excellence Workshop on Personalization and Recommender Systems in Digital Libraries*, Dublin Ireland, June 18-20.
121. Smith, R. E., & Hunt, S. D. (1978). Attributional processes and effects in promotional situations. *Journal of Consumer Research*, 5, 149-158.
122. Smith, D., Menon, S., & Sivakumar, K. (2005). Online peer and editorial recommendations, trust, and choice in virtual markets. *Journal of Interactive Marketing*, 19(3), 15-37.
123. Snyder, M., & Rothbart, M. (1971). Communicator attractiveness and opinion change. *Canadian Journal of Behavioural Science*, 3, 377-387.
124. Sproull, L., Subramani, M., Kiesler, S., Walker, J. H., & Waters, K. (1996). When the interface is a face. *Human-Computer Interaction*, 11(1), 97-124.
125. Sutcliffe, A.G., Ennis, M., Hu, J. (2000). Evaluating the Effectiveness of Visual User Interfaces for Information Retrieval. *International Journal of Human-Computer Studies*, 53, 741-763
126. Swartz, T. A. (1984). Relationship between source expertise and source similarity in an advertising context. *Journal of Advertising*, 13(2), 49-55.
127. Swearingen, K., & Sinha, R. (2001). Beyond Algorithms: An HCI Perspective on Recommender Systems. *Proceedings of the ACM SIGIR 2001 Workshop on Recommender Systems*, New Orleans, Louisiana
128. Swearingen, K. & Sinha, R. (2002). Interaction design for recommender systems. *Designing Interactive Systems 2002*. ACM, 2002.
129. Tamborini, R., & Zillmann, D. (1981). College students' perceptions of lecturers using humor. *Perceptual and Motor Skills*, 52, 427-432.
130. Taylor, P.M. (1974). An experimental study of humor and ethos. *Southern Speech Communication Journal*, 39, 359-366.
131. Tintarev, N. and Masthoff, J. (2007). Effective explanations of recommendations: User-centered design. *Proceedings of the ACM Conference on Recommender Systems*, Minneapolis, US, 153-156.
132. Tzeng, J. -Y. (2004). Toward a more civilized design: Studying the effects of computers that apologize. *International Journal of Human-Computer Studies*, 61(3), 319-345.
133. Wang, W. (2005). *Design of Trustworthy Online Recommendation Agents: Explanation Fa-*

- cilities and Decision Strategy Support. Doctoral Thesis, University of British Columbia, Vancouver.
134. Wang, W., & Benbasat, I. (2005). Trust in and adoption of online recommendation agents. *Journal of the Association for Information Systems*, 6(3), 72-101.
  135. Wang, W., & Benbasat, I. (2007). Recommendation Agents for Electronic Commerce: Effects of Explanation Facilities on Trusting Beliefs, *Journal of Management Information Systems*, 23 (4), 217-246.
  136. Wang, Y. D. & Emurian, H. H. (2005). An overview of online trust: Concepts, Elements and Implications. *Computers in Human Behavior*, 21(1), 105-125.
  137. West, P. M., Ariely, D., Bellman, S., Bradlow, E., Huber, J., Johnson, E., Kahn, B., Little, J., & Schkade, D. (1999). Agents to the rescue? *Marketing Letters*, 10 (3), 285-300
  138. Xiao, B., & Benbasat, I. (2007). E-Commerce product recommendation agents: Use, characteristics, and impact. *MIS Quarterly*, 31(1), 137-209.
  139. Yoo, K. -H. (2008). *Creating more credible and likable recommender systems*. Dissertation Proposal. Texas A&M University.
  140. Yoo, K. -H. & Gretzel, U. (2008). The influence of perceived credibility on preferences for recommender systems as sources of advice. *Information Technology & Tourism*, 10(2), 133-146.
  141. Yoo, K. -H. & Gretzel, U. (2009). The Influence of Virtual Representatives on Recommender System Evaluation, *Proceedings of the 15th Americas Conference on Information Systems*, San Francisco, California
  142. Yoon, S. N., & Lee, Z. (2004). The impact of the Web-based Product Recommendation Systems from Previous Buyers on Consumers' Purchasing Behavior. Paper presented at the tenth Americas Conference on Information Systems, New York, New York, August, 2004.

# 设计和评估推荐系统的解释

Nava Tintarevand 和 Judith Masthoff

**摘要** 本章全面描述了推荐系统的解释。我们会从评估的角度着手，也就是说，我们感兴趣的是怎样才是一个好的解释，并给出该如何更好地评估该解释的建议。我们发现解释能给推荐系统带来 7 个好处，并将它们和评估标准相联系，以及如何将它们用于实际推荐系统。我们还讨论了推荐系统的呈现方式如何影响其解释性，以及交互设计所扮演的角色。最后，本章还描述不同类型的解释，以及相关的底层算法。一些关于现有推荐系统解释的案例会贯穿本章内容。

## 15.1 简介

近年来的研究中，对于推荐系统以用户为中心的评价研究的趋势在不断增长，如文献[42]所提到的。同时人们也发现，很多推荐系统的功能如同黑盒子，不能透明地展现出推荐过程的运行进程，除了推荐系统本身，也不能提供更多的额外信息[42]。

解释可以使隐藏的原因和数据透明化。这是在亚马逊主页上有关解释的事例，例如，“购买了这个物品的用户也购买了……”。解释还能够达到其他的目标：比如，增加用户的信任和忠诚度，提高满意度，使用户可以更快、更容易地找到他们想要的，本章将区别不同的解释：如解释推荐引擎如何工作(透明度)，或解释用户喜欢或不喜欢某个物品(有效性)。有效解释可以用这种方式给出：“你可能喜欢(不喜欢)物品 A，因为……”。与上面提到的亚马逊的例子相比，这个解释没有描述物品是如何被推荐的，因此不是一个透明的解释。

15.2 节为推荐系统解释的设计和评估提供了一些指导方针。专家系统可以称作推荐系统的前身。在 15.3 节会简单地研究专家系统解释评估和推荐系统解释评估的关系。随着推荐系统的发展，可以断定过去专家系统对解释的研究可能会再度兴起。

目前为止，有两点一直未确定：如何评估解释；为什么要解释。在 15.4 节中，列出了 7 个解释标准，并说明其在过去系统中如何被衡量。这些标准也可以被理解为解释给推荐系统带来的好处，并回答了为什么要解释的问题。关于解释的有效性和透明性，从上例中可以看到这两个评价标准可能是相互排斥的。

在 15.5 节，我们认为底层推荐系统会影响对解释的评估，并根据通常用于推荐系统的评估标准(例如准确度和覆盖率)进行讨论。在整章中我们都会提及并描述一些有关解释的例子，提供一个商业和学术推荐系统实例汇总表 15.6。将以文字和图形多种形式展示解释。

此外，解释不能脱离推荐以及与用户的交互：二者相互影响，生成的解释又能反过来

影响解释的目的。在 15.6 节中讨论这些类型的设计选择。在 15.6.1 节中, 提到推荐系统不同的呈现方式, 15.6.2 节讨论用户如何与推荐系统交互。

此外, 推荐引擎的底层算法可能会影响解释类型, 尽管有开发者选择的解释可能并不反映底层的算法。有时解释的算法太复杂, 可能难以用于生产环境, 如协同过滤生成[29, 31]。在这种情况下, 开发者必须考虑权衡不同解释的目标, 如满意度(可理解)和透明度。在 15.7 节中, 阐述了最常见的解释风格和可能涉及的底层算法。最后, 在 15.8 节做了总结并讨论了未来的发展方向。

## 15.2 指引

本章的内容分为以下几个部分:

- 考虑一下你可能从解释里获得的好处, 以及为了评估相关标准的最佳衡量方法(15.4 节)。
- 注意有关解释的评估会和底层推荐引擎的作用相联系, 或者相混淆, 就像评估推荐系统通常所采用的标准所衡量的那样(15.5 节)。
- 想想推荐系统的展示和交互是如何互相影响解释的(15.6 节)。这些因素反过来会影响不同解释目标可能达成的程度。
- 最后, 但并非最不重要的, 要考虑你选择的解释类型和底层算法的关系(15.7 节)。它们能否达到解释的目标?

## 15.3 专家系统的说明

解释在智能系统上的运用并不是一个新的想法: 解释是专家系统研究的一部分[8, 32, 38, 27, 66]。这个研究主要集中于能够给出什么样的解释以及如何实现[8, 32, 38, 66]。总体来说, 在这些系统里面几乎没有对解释的评估。过去对解释的评估主要是看用户对系统[15](如其给出的结论)的认可度[67]。MYCIN 的评估是个例外, 它将系统的决策支持作为一个整体考虑[27]。相比之下, 在专家系统中, 推荐系统背后的商业目的是面向广大用户群的, 这一点既是专家系统不具备的, 又扩展出了除用户认可度外的解释评估目标。

在 20 世纪 90 年代专家系统的研究热度下降后, 推荐系统的发展重新促进了对解释的研究, 其中一个发展是由网络带来的大数据, 现在有越来越多的用户使用推荐系统, 系统也不再彼此隔离, 使代码(开源项目)和数据集得到更好的重用, 如 MovieLens[2]和 Netflix 数据集[3]。同时, 算法的优化, 特别是协同过滤已经被广泛使用和发展(参见第 4 章和第 5 章)。这些方法减轻领域依赖, 具有更大的普遍性, 尤其适用于大的稀疏数据集。解释研究重新兴起的一个标志是最近研究会上的一系列解释计算取得的成功[53, 54]。

进一步的阅读, 可以参看专家系统对三个最常见推理方法的解释综述: 基于启发式方法[36], 贝叶斯网络方法[35], 基于案例的推理[22]。

## 15.4 定义的目标

**指导方针 1:** 考虑一下你可能从解释里获得的好处, 以及为了评价相关标准的最佳衡量方法。

参照有关推荐系统解释的文献, 我们看到推荐系统所拥有的解释能力会根据不同的标准进行评估, 并确定了 7 个不同的目标。在这里提到的目标, 适用于单个物品的推荐, 即

在提供一个单一建议时给出。当作多物品推荐时，如推荐列表，标准可能是不同的，需要考虑其他因素(如多样性)。

表 15.1 明确这些目标，类似于专家系统的需求(不是评估)，参见 MYCIN[10]。在表 15.2 中，我们总结以前对推荐系统解释的评估，以及评估的标准。这些没有明确标准的工作从这个表中省略。

表 15.1 解释标准及其定义

目标	定义	目标	定义
透明度	解释系统如何工作	说服力	说服用户试用或购买
可辨认	允许用户投诉系统错误	效率	帮助用户更快地做决定
信任	增加用户的信任	满意度	增加用户的使用愉悦性
有效性	帮助用户更好地做决定		

例如，我们在 15.3 节提到对专家系统常见评估是用户认可度和整体的决策支持。用户的认可度可被定义为满意度和被说服度。如果评估对整个系统的认可度，其结果就反映了用户满意度[15]，比如，你喜欢这个程序吗？如果评估指标更能衡量用户建议或解释的接受程度，那么指标就是说服度，如文献[67]。

明确目标的区别是很重要的，即使它们会相互作用，或需要一定的权衡。但事实上，很难给出一个满足所有目标的解释，这实际上是一个取舍。例如，研究发现个性化的解释可能会增加用户的满意度，却不一定是提高有效性[61]。而且，看似有内在联系的目标实际上却不一定，例如，已经发现透明度不一定能够增加解释的信任度[20]。由于这些原因，表 15.2 中的解释可能没有通过所有标准的评估。

表 15.2 推荐系统的解释标准已经被评估出来。这些推荐系统的名称(如果有)都提到了，否则我们只标出了被推荐物品的类型。没有明确的标准规定的工作，或者没有用他们所规定解释标准评估过推荐系统的工作，都从这个表中略去了。请注意，尽管系统在几个标准方面都做了评估，但它可能不会实现所有的标准。所以，为了完整性，我们已经区分了使用相同的系统的多个研究

System(type of items)	Tra.	Scr.	Trust	Efk.	Per.	Efc.	Sat.
(Internet providers)[23]			X		X		X
(Digital cameras, notebooks computers)[49]			X				
(Digital cameras, notebooks computers)[50]			X	X			
(Music)[55]			X				
(Movies)[61]				X	X		X
<i>Adaptive Place Advisor</i> (restaurants)[59]				X		X	
<i>ACORN</i> (movies)[65]							X
<i>CHIP</i> (cultural heritage artifacts)[19]	X		X	X			
<i>CHIP</i> (cultural heritage artifacts)[20]	X		X				X
<i>iSuggest-Usability</i> (music)[30]	X			X			
<i>LIBRA</i> (books)[11]				X			
<i>MovieLens</i> (movies)[29]					X		X
<i>Movieexplain</i> (movies)[58]				X			X
<i>myCameraAdvisor</i> [63]			X				
<i>Qwikshop</i> (digital cameras)[39]				X		X	
<i>SASY</i> (e. g. holidays)[21]	X	X					X
<i>Tagsplanations</i> (movies)[62]	X			X			



解释的类型依赖于某个推荐系统设计的标准。例如,当建立一个售书系统时,用户的信任是最重要的一个标准,因为它会增加用户的忠诚度和销售额。对于视频系统,用户满意度比有效性更加重要。即用户喜欢某个服务比呈现给他们最可用的展示要重要。

此外,解释一些属性可能有助于实现多个目标。例如,解释的明白程度有助于提高用户的信任以及满意度。

在本章节中描述了有关解释的7个标准,并且建议了基于解释器以前评估的评估方法,或者是对已有方法如何适应评估推荐系统中的解释功能给出了建议。

#### 15.4.1 系统如何工作:透明性

在华尔街杂志上有篇题为“如果 TiVo 认为你是同性恋,本文来解释它的原理”的有趣文章,描述了视频播放器根据观众过去看过的节目,假设它的主人会喜欢而向他播放节目,做出不相关的选择,因而给用户带来的困扰[69]。例如,有位 Iwanyk 先生,怀疑他的 TiVo 认为他是同性恋,因为 TiVo 不断播放同性恋相关题材的节目。该用户肯定需要合理的解释。

一个解释可以说清楚推荐是如何被选择的。在专家系统中,如在医疗决策领域,透明度同样重要[10]。透明性或系统状态的可视性是一个确定的可用性原则[44],同时也是用户推荐系统研究的一个重点[55]。

Vig 等在文献[62]中区分了透明度和理由。虽然透明度要求对于如何选择推荐结果以及推荐系统如何工作应该给出诚实的解释,但理由可以是描述性的,并且与推荐算法解耦。作者举出几个选择理由,而不是透明度的原因。例如,一些算法难以解释(例如,隐语义分析中的因子是潜在的,没有清晰的解释),系统设计师需要保护商业秘密,并且渴望在解释的设计里获取更大的自由。

Cramer 等研究艺术品推荐系统里透明度在其他评估准则上的作用,如信任度、说服度和满意度[19, 20]。透明度的评估准则是:感知并理解系统如何工作[20]。而实际的感知和理解是基于用户采访和调查问卷。

尽管透明度和被理解(15.4.2节)、信任度(15.4.3节)是耦合在一起的,但我们将会看到,它们的标准还是不同的。

#### 15.4.2 允许用户告诉系统它是错误的:被理解

解释可能有助于隔离和纠正被误导的假设或步骤。当系统收集和解释背景的信息时,如同在 TiVo 的案例中,对于用户来说推理的可行性变得更加重要。透明度之后的第二步,就是要让用户能够纠正推论,或是让系统能够被理解[21]。解释应该是周期的一部分,用户明白在系统里是怎么回事,在系统中提出建议并且能够在需要时纠正系统的假设[56]。可理解性与用户控制[44]的确定的有用性原则相关联。图 15.1 为一个可理解的假期推荐的例子。

虽然可理解与透明度密切相关,但应该被唯一地标识出来(见表 15.3)。表 15.4 中的解释是可以理解的,但不(完全)透明,即使它们提供某种形式的理由。例如,在该表中,没有东西标明的层是基于贝叶斯分类器的。在这种情况下,我们可以想象一个用户试图理解一个推荐系统,并设法通过修改评分来改变推荐结果,但他们仍然不明白到底系统底层发生了什么。

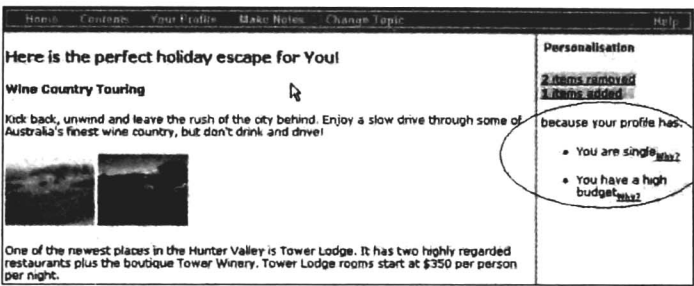


图 15.1 可理解的假期推荐[21]。该解释位于画圈的区域，用户的属性可以通过“原因”链接来访问

表 15.3 关键词风格解释[11]，该推荐的解释是按照已经使用描述物品关键词，以及有关联的高评分物品。数量(Count)描述关键词出现在描述物品中的次数，优点(Strength)描述该关键字预测物品喜好的程度

Word	Count	Strength	Explain
HEART	2	96.14	Explain
BEAUTIFUL	1	17.07	Explain
MOTHER	3	11.55	Explain
READ	14	10.63	Explain
STORY	16	9.12	Explain

表 15.4 在点击表 15.3 中“解释”之后将会展现关键字“优点”更多的解释。实际上，优点的概率度量的是关键字出现在积极的物品中比出现在消极的物品中的程度。其基于用户之前的积极评分，以及描述这些物品的关键词的数量

Title	Author	Rating	Count
Hunchback of Notre Dame	Victor Hugo, Walter J. Cobb	10	11
Till We Have Faces: A Myth Retold	C. S. Lewis, Fritz Eichenberg	10	10
The Picture of Dorian Gray	Oscar Wilde, Isobel Murray	8	5

Czarkowski 发现用户本身不容易理解这些，所以需要额外的努力来使得理解更加可视化[21]。此外，让用户通过改变个性化结果来理解系统会更容易(如“改变个性化结果：使 4：30~5：30 的进度表中只包含实时节目”)。他们的评估包括像任务正确性这样的衡量标准，以及如果用户可以表达一些信息，什么信息被用作对他们的推荐。他们明白，推荐结果是基于他们的个人属性，即系统存储的他们的个人资料。信息中包含他们自愿的信息，并且可以通过改变它们来控制个性化[21]。

15.4.3 增加用户对系统上的信任：信任度

信任有时候是和透明度相关联的，以往的研究表明，透明度和用户与推荐系统的交互会增加用户对系统的信任[23, 55]。用户会更宽容和信任，如果他们了解一个不理想的推荐结果是如何产生的。信任也依赖于推荐算法的精确性[41]。对用户的信任研究表明用户打算对值得信赖的推荐系统做出反馈[16]。然而，我们注意到，有一种情况下，透明度和信任是不相关的[20]。

我们并不认为解释完全可以弥补推荐的不好，但很好的解释可以帮助用户做出更好的决定(见 15.4.5 节效果)。用户也会喜欢系统的透明性以及承认对于特殊推荐没有信心。

此外，一个推荐系统的界面设计可能会影响其信誉。在确定网页的可信度因素的研究里，用户评论占比最大(46.1%)的是网络的整体视觉设计：包括布局、字体、字体大小和颜色方案[25]。同样作者照片的呈现也显著影响网站的可信度[24]。因此，推荐的准确性和透明性关系到评估的可信度，同时设计也是评估中需要考虑的一个重要因素。

问卷可用于确定用户对系统的信任度。关于信任问卷的概述可以在文献[45]中找到，其中也建议并验证了五维的信任级别。请注意，这个验证的目的是利用信誉担保产品，而不是针对某个特定领域进行的。可能需要更多的验证将这种信任级别适应到某个特定的推荐领域。

推荐系统的信任模型在文献[16, 50]中提出，这些调研的问卷调查中考虑的因素包括：是否愿意对系统反馈，是否提升了效率。同样文献[63]提出用户信任，但是关注与信念相关的信任，例如，一个虚拟顾问的能力感、仁慈、真实。虽然问卷调查可以非常集中，但有用户言行不一的顾虑。在这些情况下，隐含的测试(虽然不太注重)可能揭示一些因素，这些因素都是显性测试目的所不能获得的。

忠诚，一种令人满意的信任的副产品，可能是这样的一种隐式方法。有研究在用户如何影响像忠诚度这样因素的方面，比较了提取用户偏好的不同界面[41]。它使用用户登录或交互的次数来衡量其忠诚度。在其他方面，研究发现是否允许用户评估物品会影响其忠诚度。亚马逊是一种保守的推荐，主要通过推荐用户熟悉的物品，增加了用户的信任和自身的销售[57]。我们鼓励那些想要了解更多有关信任度的读者去阅读第 20 章，会专门讨论该话题。

15.4.4 说服用户尝试或购买：说服力

解释可能增加用户对系统或特定推荐[29]的接受度。之所以定义为说服力，是因为这是对系统的效益，而不是用户。

文献[20]按照在最终六个最喜爱的选择中被选中的推荐数量评估推荐物品的接受性。在协同过滤和基于评分的电影推荐系统研究中，参与者被赋予不同的解释界面(图 15.2)[29]。该研究直接调查用户面对 21 种不同的解释界面时会有多可能看一部电影(识别出像遗漏标题这样的特征)。因此，说服力是在七点 Likert 级别上的数值评分。

此外，如果物品的评估发生了变化，也就是用户在收到解释后对物品的评分有所不同，也有可能衡量出来。事实已经表明，用户可以被操纵给出接近系统的预测的评分[18]。这项研究是在电影租赁低投入的领域，在投入成本领域，如相机，用户就可

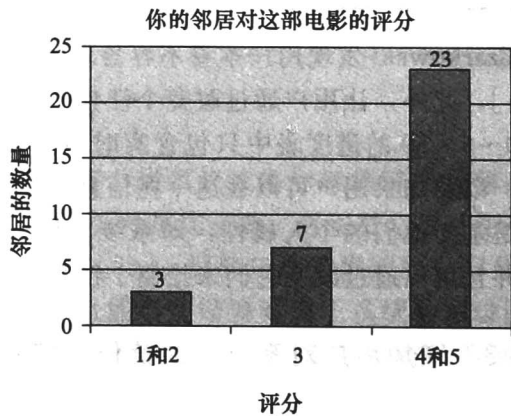


图 15.2 评估说服力的 21 个界面中的 1 个界面——直方图总结类似用户的评级(相邻的)的推荐物品，从 1 至 5 按好(5, 4)、中性(3)、差(2, 1)分级[29]

能不那么容易受到不正确预测的影响<sup>①</sup>。也需要考虑到,一旦用户发现他们已经尝试或购买一些他们不想要的,太多的说服力可能会适得其反。

衡量说服力有几种不同的方法。例如,可以测量在两种评分间的区别:一个是以前的评分,另一个是带有解释后,对相同物品的重新评分[18]。另一个可能性是对比另一个没有解释的系统,有多少用户尝试或购买物品。这些指标也可以用常用的电子商务的“转化率”概念。操作上定义为参观者采取行动的百分比。想更深入地讨论推荐系统说服力的读者可以参考第 14 章。

#### 15.4.5 帮助用户充分地决策:有效性

除了简单让用户试用或购买一个物品,解释也可以帮助用户做出更好的决策。有效性根据定义来看高度依赖推荐算法的准确性。一个有效的分析能帮助用户根据自己的喜好评估推荐物品的质量。这将帮助他们识别有用物品而丢弃无关选项。例如,一个有效的图书推荐系统可以帮助用户购买他们最终喜欢的书籍。Bilgic 和 Mooney 强调一种测量系统来帮助用户基于解释做出对推荐结果的准确决策,如图 15.3 和表 15.3、表 15.4 和表 15.5 [11]。有效的解释也适合于新类型产品,或新用户,因而帮助他们理解全部选项[23, 49]。

Vig 等衡量了已感知的有效性:“解释帮助确定我会喜欢这部电影。”[62]。解释的有效性也可以通过用户在消费推荐物品之前和之后喜爱程度的差异来计算。例如,在以前的研究中,用户两次评估一本书,一次是在得到解释时,另一次是在读它时[11]。如果他们对这本书的意见没有多大变化,该系统被认为是有效的。这项研究探讨了整个推荐过程,包括解释,对于有效性的影响。相同的衡量方法也被用于评估个性化解释(独立于推荐系统)是否能在电影领域提高有效性[61]。

虽然这种衡量方法考虑了评分前后的差异,它没有讨论过低估计会产生的效果。如果公开一个物品,用户的评估减少,那么他们的初始评分是一个高估的信息。同样,如果他们的评估在物品公开后增加,那初始评分被低估。在我们的工作中发现,用户认为高估的有效性比低估的有效性少,它存在于各种不同领域之间。特别是过高估计在高投入领域比低投入领域更为严重。此外,对感知效果影响的强度取决于预测误差发生的规模大小[60]。

另一种衡量解释有效性的方法是,评测同样的系统在有或没有解释机制下,评估收到解释的受试者是否能最终找到更适合他们个人品位的物品[19]。

其他评估解释有效性的工作采用了来自营销领域的衡量方法[28],以找到一个可能最好的物品为目的(而不是以上的“足够好的物品”)[17]。参与者与系统相互作用,直到他们找到想要购买的物品。然后他们被给予机会去检查整个目录以及改变他们所选物品的选项。一部分找到好物品的参与者会被用来测量效果,这一部分被选的参与者是通过比较数据库中所有的参与者选择出来的。因此,使用这种度量,低分数代表高有效性。

有效性这个标准与精度测量最为相关,如准确率和召回率[19, 58, 59]。对于那些快

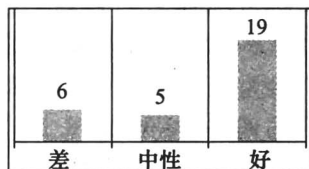


图 15.3 邻居风格解释。直方图总结了类似用户的评级(相邻的)的推荐物品,从 1~5 按好(5, 4),中性(3),差(2, 1)[29]。与图 15.2 类似,该研究也是有意强调并且突出劝说性和解释之间的区别

① 文献[60]中记录了他们发现,相对低成本领域,在高成本领域不正确的过高估计的用处更少。

消费物品的系统，这些衡量标准可以转化为分别识别相关物品和丢弃不相关选项。例如，有一种替代“准确率”的衡量标准是，匹配用户兴趣的概念数量，与用户个人信息中的概念总数之比[19]。

#### 15.4.6 帮助用户快速制定决策：效率

解释可以使用户更快地决定哪些建议物品对他们来说是最好的。效率是另一个确定的可用性原则，即如何快速地执行任务[44]。如果推荐系统要做的事情是在大海捞针，那么这个准则是所有推荐系统文献中都要解决的问题。

帮助用户理解相互矛盾的选项之间的关系有助于提高效率。[39, 43, 49]采用所谓的评价算法就能很好地生成解释，该算法是在物品属性之间进行权衡的基于知识算法的一个子集。如在数码相机领域，选项相互冲突的例子可能是：“低内存、低分辨率、便宜”[39]。这样如果用户满足于一个配置为低内存、低分辨率的相机，可以通过这个查询修正迅速找到一个便宜的相机。基于评价的推荐系统更多的细节在第13章有详细的描述。

效率经常被用来评估会话式推荐系统，该系统可以使用户与推荐系统进行交互，同时细化他们的偏好(参见15.6.2节)。在这些系统中，可以在对话中看到隐含的解释。效率可以通过用户在交互完成时找到满意物品的情况下所花费的总时间和交互次数来计算[59]。根据效率提升来评估解释的作用并不限于会话式系统。举个例子，Pu和Chen不仅比较了两个解释界面的完成时间，并且将完成时间衡量为参与者在界面中定位到所需产品花费的时间[49]。

当没有合适的物品呈现时，对效率的度量方式包括检查解释的次数和主动修正行为的次数[23, 52]。向用户显示所有可能的推荐结果及相关解释是明智的，这样可以使用户通过解释详细地浏览物品。在更高效的系统中，用户只需要检查少量解释。修正行为来自用户的反馈，用于改变用户看到的推荐物品，如可理解性(15.4.2节)所总结的。用户反馈或者行为修正的实例可以在15.6.2节中找到。

#### 15.4.7 使系统的应用愉悦：满意度

人们发现解释可以提高用户对推荐系统的满意度或接受程度[23, 29, 55]。对物品进行详细的描述既利于用户的体验[60]，又使得推荐系统方便使用[55]。另外，很多商业推荐系统来源于娱乐领域，如表15.6所示，在这种情况下，所有额外的机制都应致力于提高用户满意度。表15.4给出了在满意度维度上对解释评估的例子。

当衡量满意度的时候，一方面可以直接咨询用户该系统是否是易用的[15]，另一方面判断用户是否喜欢这些解释[60]。满意度也能间接地通过用户忠诚度衡量(详细部分见15.4.3节)[41, 23]，或者通过判断该系统是否能用来完成搜索任务[20]。

在衡量解释的满意度时，区分用户是对推荐的过程满意还是对推荐的产品<sup>①</sup>满意很重要[20, 23]。衡量推荐过程满意度的一种定性方法是进行可用性测试，比如，用有声思维法记录用户执行某个任务[37]。

在以下这个例子中，参与者在执行一个任务，比如，找到一个满意的物品，他们使用系统描述他们的整个经历，包括看了什么、想了什么、做了什么和感觉如何。系统不以任

① 这里指的是整个推荐过程，包括解释。然而在15.5节，我们会强调指出推荐系统中解释的评估很少完全独立于基本的推荐过程。



何解释或其他方式影响用户，使其客观地记录一切。视频和语音录音也可以用于再次查看会话或者作为记忆辅助物。在这种情况下，有可能发现可用性问题的，甚至采用定量的度量标准，如正面评论和负面评论的比例，评估者感到沮丧的次数，评估者感到高兴的次数，评论者在什么地方遇到了可能存在的问题及遇到的次数等。

我们认为用户也会满意将两种标准混合起来的高效解释系统。然而，一个能帮助用户做决策的系统，可能会降低系统整体的满意度(如需要大量对用户的认知工作)。幸运的是，这两个标准可以通过不同的指标进行衡量。

15.5 评估解释在推荐系统的作用

指导方针 2：采用通常用于评估推荐系统的标准来衡量，对于解释的评估会与底层推荐引擎的功能有某种关联，甚至会相互混淆。

我们发现 7 个标准可以衡量推荐系统的解释效果，这里给出了可以用于评估的建议。在某种程度上，这些评估标准可以评估用于解释的组件，评估整个系统似乎也是合理的。这种情况下，我们会衡量一般系统的可用性和准确性，这些系统同时依赖于推荐算法和解释组件。因此，在这一部分中，我们描述推荐系统和解释标准之间的交互时，常用的参数有精准度、学习效率、覆盖率、新颖性/惊喜度和接受度。

该推荐结果的理由如下：

- 这类连接的网络空间是可用的；
- 对您来说，这个包是可以连接到您的任意一个位置；
- 网络连接的月消费超出您的预算。

图 15.4 一个互联网服务提供商的解释，描述了提供物品的用户需求：“该结果由以下原因得出……” [23]

Prediction for *Sixth Sense, The*(1999)

In Doph	MovieLens
Prediction	Confidence
★★★★★	★★★★★

图 15.5 推荐结果的可信度 [29]——该电影是强烈推荐的，并且有大量的信息支撑着该推荐结果

15.5.1 精准度

精准度代表推荐系统在预测方面的能力，但是也会与解释有关。例如，就精准度与透明度的关系而言，Cramer 等发现透明度会改变用户的行为，最终降低精准度 [19]。

推荐系统自身的可信度与精准度也有关系，并且能在解释中体现。例 15.5 中可信度的展示旨在帮助用户理解精准度。这些可以用来解释不好的推荐结果，如那些没有足够的信息来形成的推荐结果。进一步有关可信度方面工作的介绍可以参见文献 [40]。

解释能够帮助用户了解如何对待一个特定的推荐结果，可能提供的额外信息能帮助用户做出明智的选择并提高效率。在精准度很低的情况下，会导致丢失好的推荐结果而坏结果增加的风险，然而，解释可以降低这种风险。通过帮助用户识别推荐结果的好坏，整体推荐系统的准确度也会提升。

15.5.2 学习效率

学习效率标志着推荐系统对用户偏好变化的敏感程度。当一个用户希望推荐系统能快速学习到他们的喜好，对待短期兴趣也能像长期兴趣一样敏感时，学习效率会影响用户的满意度。解释可以通过暗示用户系统注意到了其偏好的变化来提高满意度。例如，系统可以给一个给定的变量标记一个值，使得它因为改变而越来越接近阈值，但还没达到该阈

值。系统还可以使用户了解它是如何学习并改变偏好的(透明度),甚至可以允许用户修改他们的旧偏好(可理解性)。解释机制可以请求一些信息帮助系统更快地学习或者改变,如可以询问用户对电影类型的喜好是否从动作片转移到了喜剧片。

### 15.5.3 覆盖度

覆盖度标识着推荐系统推荐结果的覆盖能力。解释能够帮助用户了解他们的搜索空间大小。在探索搜索空间中,解释能帮助用户选择信息丰富的物品,这能增加部分物品或者特征的重叠(与稀疏性相对)。最终,这会为潜在的推荐增加总体覆盖度。理解剩下的搜索选项与透明度标准有关:推荐系统可以解释为什么某些物品没有被推荐。或者不可能,或者难以检索该物品(例如,物品在基于内容的系统中有一个特别的属性,或者在协同过滤系统中没有足够的点击量)。此外,推荐系统可能基于用户对该物品不感兴趣的基础上设计函数(例如,如果用户的需求在基于内容的系统中太狭窄,或者在协同过滤系统中属于小众口味)。一个解释能说明为什么该物品对推荐系统是不可用的,甚至能说明如何补救这种情况并且允许用户改变他们的喜好(可理解性)。

覆盖度可能会影响有效性的解释标准的评估。例如,一个用户不是要找到一个足够好的结果,而是要找到一个最好的结果,那么覆盖度需要确保这最好的结果在推荐结果中。取决于检索物品集合的速度,覆盖度也会影响效率。

### 15.5.4 接受度

人们可能会将对一个系统的接受度或满意度,与其他类型的满意度相混淆。如果一个用户对一个有解释组件的推荐系统满意,我们仍然不清楚用户是满意解释组件,还是满意推荐系统,或者满意整体设计风格和视觉吸引力。对系统的满意度与推荐精准度紧密相连,甚至与新颖性和多样性相连,从某种意义上说,要有足够好的推荐结果才能满足用户需求。虽然解释能帮助系统提高满意度,但是他们不能替代一个好的精准度。事实上,这适用于所有的解释标准。一个符合满意度标准的解释例子如下:“请完全向我展示,在为您做出精准推荐之前,我需要学习更多关于你的偏好。”

## 15.6 用推荐设计展示与互动

**指导方针 3:** 思考以下几个问题:如何展示推荐结果,如何展示交互模型,如何展示互相的影响以及如何展示解释。这些因素将影响解释达到目标的程度。

展示推荐结果的方式可能会影响交互模型,而该模型可以提取到用户的偏好情况。同样,以上两个因素会影响解释的生成。换句话说,一些解释的生成可能更适合特定的解释标准。第 16 章同样讨论了面向个性化推荐系统的一个互补的评估框架(如在第 13 章中描述了评价机制),重点在于设计推荐的展示和交互的模型。例如,一个指导方针描述的:“同时展示一个搜索结果或者推荐一个物品可以简单并适合展示在一个小的显示器上;然而,这样却可能让用户停留在更长的交互会话中,或者仅允许他们得到相对低的决策精度。”(详见指导方针 9。)

### 15.6.1 展示推荐

我们总结一下在本书中出现过的推荐展现方式。用户图形界面的外观有很多种方式,

提供推荐的实际结果也各不相同。我们定义以下几种组织推荐结果展现形式的类别：

- **最优物品。**或许展示推荐的最简单的方式就是为用户提供最优物品。例如，“您曾经看过很多运动相关物品，尤其是足球类的。这个是世界杯期间最流行且最新的物品。”
- **最优 N 物品。**系统也可能一次展示多个物品。“你看了很多足球类和科技类物品。你可能会喜欢看当地的足球结果和当天的科技新产品。”注意这个系统能解释物品之间的关系，它也可以解释每个单项背后的理由。
- **最优项相似物品。**一旦用户表现出了一个或多个偏好，推荐系统能够提供类似的物品。如“你可能还喜欢……查尔斯·狄更斯的《雾都孤儿》。”
- **为所有的物品预测并评级。**系统允许用户浏览所有可用的物品，而不是强迫用户做出选择。推荐是通过为每一项预测评分后被推出的(分值从 0 到 5)。用户可以查询如当地曲棍球这样的物品为什么被预测了一个很低的分值。推荐系统最后生成如下的解释：“这虽然是一项运动，但和曲棍球相关，你可能并不喜欢它。”
- **结构概览。**推荐系统能提供一种结构，实现在物品之间进行切换的功能[49, 68]。结构概览的好处是用户可以看到物品之间的比较，如果当前推荐结果不能满足需求，其他结果仍然可用。

### 15.6.2 与推荐系统交互

用户可以通过不同的方法向推荐系统提供信息。交互可以区分为会话式系统和单向推荐系统。他们允许用户通过一个扩展对话框提交复杂的需求[51]，而不是每个用户的交互独立于以前的历史记录单独处理。

我们提供当前应用程序的例子<sup>⊖</sup>，用于拓展四个方面的建议[26]。注意尽管有更多的方法来获取用户偏好，如通过使用数据[46]或者人口分布[6]，本节的重点在于用户的显式反馈。

- **用户指定他们的需求。**用户能够用简单的英语通过对话来指定他们对偏好的需求[43, 64]。这样的对话不用于用户以前的兴趣上，也不用于直接的解释上。也就是说，没有词句可以生成推荐理由。然而，它是间接的，通过重复满足用户的需求。
- **用户请求换一换。**一个更直接的办法是允许用户显式地评价推荐系统(见第 13 章的评价的改进)，例如，使用结构化概览(见 15.6.1 节)。一个这样的系统可以解释物品和剩余物品之间的区别[39]。
- **用户给物品评分。**为了改变他们得到的推荐结果的类型，用户可能想纠正预测评分，或者改变过去的一个评分。表 15.5 中基于解释的影响表明，评分标题是影响推荐书籍的最重要因素[11]。
- **用户提供他们的意见。**常见的可用性原则比从内部描述更容易使人了解推荐物品。例如，如果一个用户想了解更多类似的物品或者他们曾经已经看过该物品，则可以指定这个物品在他们看来是否有趣[12, 57]。
- **混合交互界面。**推荐系统也可以结合不同类型的交互[17, 41]。

⊖ 五分之一的关于混合交互接口的内容添加到了这个原始列表中。

表 15.5 基于解释的影响表明评分标题对推荐数据影响力最大。尽管这个系统不允许用户修改以前的评级或者影响程度，但是可以直接在解释界面更改评分。然而注意一点，修改影响程度会比较困难，因为该值是通过计算得来的：任何修改都会干扰常规的推荐算法函数[11]

书名	您的评分(总分 5 分)	影响力(总分 100 分)
Of Mice and Men	4	54
1984	4	50
Till We Have Faces: A Myth Retold	5	50
Crime and Punishment	4	46
The Gambler	5	11

15.7 解释风格

指导方针 4：考虑底层算法和你选择生成解释的类型之间的关系。你生成的解释能帮你达到解释效果吗？

在该部分我们描述基于特定底层算法的解释方式或者不同的“解释风格”。注意一点，无论推荐结果是如何被检索或者计算得来的，解释风格可能与特定算法无关。换句话说，解释风格可能会，也可能不会反映出计算出推荐结果的底层算法。推荐是如何被检索出来的与解释风格是如何生成的两者之间经常有分歧。因此，这种类型的解释与透明度的目标可能不一致，但是或许会支持其他的一些解释性目标。

透明度并不是决定解释风格的唯一标准。例如，对于一个特定的推荐系统，相比效果好的基于评价的解释，用户更喜欢基于内容的解释。然而，在解释风格和解释目标之间几乎没有对比。只有 Hingston[30]比较了不同受算法诱导的解释风格之间的可理解性和可读性，在这些例子中，解释也直接受底层算法影响。然而其他研究基于不同的解释目标研究了不同的解释界面[20, 29, 61]。

尽管推荐引擎的底层算法在一定程度上会影响解释类型的生成。表 15.6 总结了最常用的解释风格(案例、协同、内容、会话、人口统计、基于知识/自然语言)并附有例子，在这部分我们描述每一种风格，包括他们的输入、处理过程及生成的解释。在商业系统中这些信息都不是公开的，我们提供一些学术上的猜测。虽然会话式系统也包含在这个表中，但是我们认为会话系统比某个特定算法具有更多的交互样式。

表 15.6 商业和学术系统的解释实例，按照解释风格排序(案例、协同、内容、会话、人口统计、基于知识/自然语言)

系统	解释实例	解释风格
iSuggest-Usability[30]	见图 15.8	案例
LoveFilm.com	“因为你选择过电影 A 或者给 A 打过高的评分”	案例
LibraryThing.com	“有用户 X 推荐数据 A”	案例
Netflix.com	用户历史上有高评分电影的相似电影集合	案例
Amazon.com	“买了该物品的用户还买了……”	协同
LIBRA[11]	关键词风格(见表 15.3 和表 15.4)；邻居风格(见表 15.3)；影响力风格(见表 15.5)	协同
MovieLens[29]	邻近者(见表 15.2)和信任度(见表 15.5)直方图	协同

(续)

系统	解释实例	解释风格
Amazon.com	“因为您告诉我们您有书籍 A”	内容
CHIP[20]	“为什么为您推荐了 <i>The Tailor's Workshop</i> ? 因为它与您喜欢的以下艺术作品在主题上相似: * <i>Everyday Life</i> * <i>Clothes...</i> ”	内容
Movieexplain[58]	见表 15.7	内容
MovieLens: “Tagsplanations”[62]	标签按照相关性或者偏好排序	内容
News Dude [12]	“这个故事相关性分值很高/低, 因为它包含以下关键词: f1, f2, 和 f3.”	内容
OkCupid.com	图表从“更内向”等维度比较两个用户; 从如何回答不同问题的方式比较用户	内容
Pandora.com	“根据您以前告诉我们的信息, 我们推荐该物品因为它有悠闲的节奏这个特征……”	内容
Adaptive place Advisor [59]	“你想去哪里吃饭?”“或许去一个便宜点的印度餐馆”	会话
ACORN[65]	“你喜欢什么类型的电影?”“我喜欢惊悚片。”	会话
INTRIGUE[6]	“它比较适合小孩, 要求知识背景低, 需要认真和迅速地浏览。适合您自己因为它有很高的历史价值。”	人口统计
Qwikshop[39]	“低内存, 低分辨率, 便宜”	基于知识/自然语言
SASY[21]	“……因为资料显示: 您是单身, 有较高的预算”(见表 15.1)	基于知识/自然语言
Top Case[43]	“574 号物品与您的请求仅价格不同, 并且无论从交通、持续时间还是住宿都是您喜欢的最优情况”	基于知识/自然语言
(Internet Provider)[23]	“该解决方案被选中的原因如下: 对该类型的连接网络空间是可用的”(见表 15.4)	基于知识/自然语言
“Organizational Structure” [49]	结构化概览: “我们推荐以下产品是因为: 他们更便宜且轻便, 但是速度比较慢。”	基于知识/自然语言
myCameraAdvisor[63]	例如, “……相机拍摄距离越远, 价钱越贵……”	基于知识/自然语言

在接下来的部分, 我们将给出更多的例子讲述解释风格是如何被像 Burke[13]分类一样的算法影响的。对每一个例子, 我们还会提及如何展示推荐结果和选择交互模型。

为了清晰地描述推荐系统和解释组件之间的接口, 我们使用文献[13]中提到的部分符号:  $U$  代表一组用户的偏好,  $u \in U$  代表需要被生成推荐结果的用户。  $I$  代表被推荐的物品,  $i \in I$  代表我们要预测用户对该物品的喜好值。

15.7.1 基于协同风格

基于协同风格的解释认为推荐引擎的输入是用户  $u$  对物品  $I$  的评分。这个评分用来识别给  $I$  的评分与  $u$  相似的用户集合。这些相似的用户经常被称为邻居, 最近邻方法用来计算用户之间相似度。这样, 对推荐物品的预估就推广为邻居对物品  $I$  的评分。

商业上, 基于协同风格解释的最著名的应用是亚马逊网站: “买了此商品的用户还买了……”。这个解释的前提是用户正在浏览一本他们感兴趣的物品。它旨在找出相似的用户集合(这些用户都买过这个物品), 并检索和推荐出这些用户买过的其他物品。推荐以与最优项相似的形式展示。此外, 解释假设存在交互模型, 凭借用户的购买这种隐性行为来评分。



Herlocker 等提出了 21 种使用文本和图形界面的解释界面[29]。这些界面的内容和风格均不相同，但是大部分解释都直接来自邻居的信息。以图 15.2 为例，它显示了邻居如何给推荐的物品进行评分，以“好”“可以”和“不好”的条状图聚成不同的列。我们看到这个解释是为特定的推荐物品和特定的交互模型提供的：这是单个推荐结果(或者是最优项，或者是最优  $N$  项中的一个)，并且假设用户提供了对物品的评分。

15.7.2 基于内容风格

基于内容风格的解释认为推荐引擎的输入是用户  $u$ (子集)对物品  $I$  的评分。这些评分用来生成一个适合用户  $u$  评分行为的分类器，再通过这个分类器发现推荐物品  $I$ 。对推荐结果的预估在于该结果对分类器的适应程度，如它是否与其他高评分物品相似。

简单来说，基于内容的算法是基于物品的属性而不是用户的评分来计算物品之间的相似度的。同样，基于内容风格的解释也是基于物品的属性产生的。例如，[58]证明电影推荐结果是根据所提供的用户最喜欢的演员得到的(见表 15.7)。当底层算法是一个基于协同与基于内容混合的方法时，解释风格建议根据高评分电影出现的特征计算电影之间的相似度。他们为当前用户选择一些可能更合适的推荐结果和解释(top- $N$  项)，如果用户愿意根据解释信息进行选择(例如，感觉像是看电影 Harrison Ford 是因为 Bruce Willis 主演的)。交互模型基于对物品的评分生成。

[62]提出的一个与领域无关的方法是基于用户指定的关键词及标签计算物品之间的相似度。在该项研究中用到的解释采用关键词语物品之间的关系(标签相关性)和标签与用户之间的关系(标签属性)进行推荐(见表 15.7)。为一个用户定义的标签属性或者相关的标签，可以作为基于内容解释的形式，因为它通过该用户给出评分的电影集合包含的标签进行加权平均计算得来的。标签相关性或相关的关键词用于推荐物品，即用户标签属性和他们对电影喜好的相关性，这些电影也存在相关的标签属性。在这个例子中，为用户推荐一个最优项可以允许用户看到很多相关标签结果。交互模型是基于数值评分的。

表 15.7 Movieexplain 解释的例子，使用演员等特征(这些特征在该部电影出来前被用户打了高分)来说明推荐结果

推荐电影名称	理由参与者	谁出现了
Indiana Jones and the Last Crusade(1989)	Ford, Harrison	你曾评了 5 部电影
Die Hard 2(1990)	Willis, Bruce	你曾评了 2 部电影

Pandora 商业系统的推荐结果是根据音乐的节奏和音调等属性计算得来的。这些特征通过用户对歌曲的评分生成。图 15.6 展示了这样一个例子[1]。这里为用户提供了一首歌曲(最优项)，用户可以通过“顶”和“踩”来表达他们的意见，这些意见可以看作数值评分。

15.7.3 基于案例风格

解释也可以忽略重要属性而聚焦于相似度进行推荐。因此推荐物品被认为是比较得出的结果，所以解释是基于案例风格的。CBR 系统与其他系统对推荐算法的重

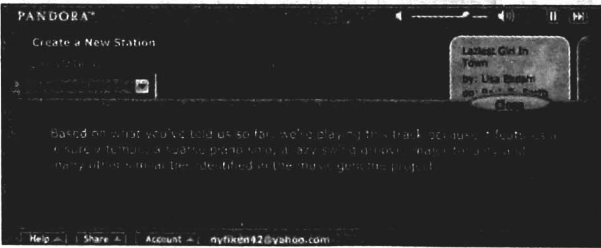


图 15.6 Pandora 的解释：“基于你之前告诉我们的信息，我们推荐这个是因为它有悠闲的节奏……”

视程度是非常不一样的。例如，推荐结果‘FINDME’[14]是基于评价得到的，排序[5]是依据相似用户对他们旅行计划的描述。虽然 CBR 系统也使用了不同的方法来展示他们的解释，但是回想本节其他部分，该节描述的其他解释风格重点在于风格而不是底层算法。因此，这些系统理论上可以有一个基于案例风格的解释。

事实上，在本章中我们已经看到了基于案例风格的解释。在图 15.5 中出现的“基于影响力风格的解释”[11]。推荐物品的影响力是通过计算在有或没有该物品的情况下推荐得分的变化。这种情况下，推荐将在基于交互的评分的前提下，以最优项的方式展示。另一项研究是计算推荐物品之间的相似度<sup>⊖</sup>，并且在“基于实例”的解释中将这些相似物品为最优项推荐的理由(见图 15.8)[30]。

### 15.7.4 基于知识/自然语言风格

基于知识和自然语言风格的解释认为推荐引擎的输入是对用户  $U$  的需求或者兴趣的描述信息。推荐引擎推断物品  $i$  与用户  $U$  的需求之间的匹配程度。一个基于知识的推荐系统考虑如何用相机的属性(如内存、分辨率和价格)反映可用属性及用户偏好[39]。他们的系统可能用以下方式解释相机的推荐：“更少的内存、更低的分辨率、更便宜。”这里的推荐结果通过结构化描述展示它的竞争性，并且交互模型可以使用户改变他们对推荐物品的要求。

相类似，在系统描述[43]中用户逐渐制订(或修改)他们的偏好直到推荐出最满意的结果。这个系统可以为推荐结果“Case 574”生成如下的解释：“最优项：Case 574 与您的查询条件仅在价格上不同，并且无论是交通、时间还是你喜欢的类型都是最好的结果”。

知识之间存在一定程度的重叠，基于内容风格(见 15.7.2 节)和基于案例风格(见 15.7.3 节)可以由任意类型的算法实现细节得到。

### 15.7.5 基于人口统计风格

基于人口统计学的解释，认为推荐引擎的输入是对用户  $U$  的人口统计信息。基于此，推荐算法会识别在人口分布上与  $U$  相似的用户集合。对推荐物品  $I$  的预测值源于相似用户集合对  $I$  的评分和这些用户与  $U$  的相似程度。

测试大量的基于人口统计过滤的系统如[6, 34, 48]，我们只发现存在一个解释群：“对孩子是有吸引力的，它对知识背景要求低，它需要认真且快速浏览。对您来说它是有

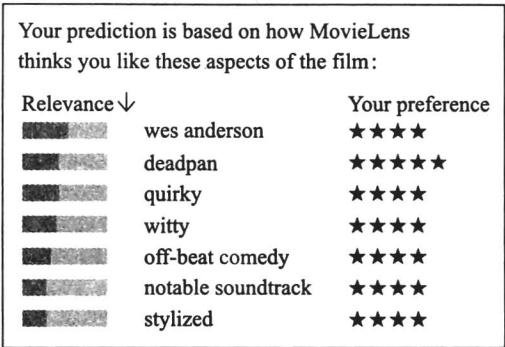


图 15.7 通过标签属性和相关性计算标签作用，但是通过标签相关性进行排序

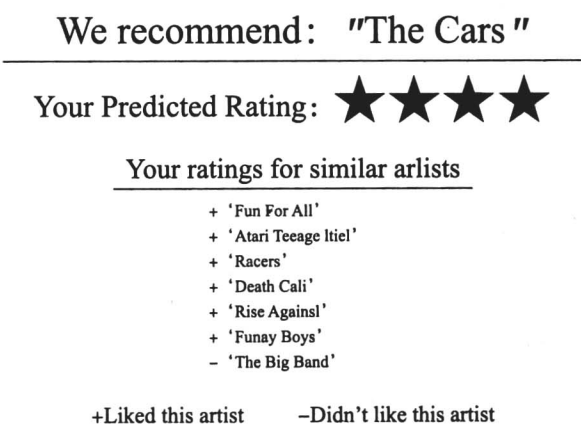


图 15.8 通过例子学习，或者通过案例的推理[30]

⊖ 作者并没有指定采用了哪种相似度标准，尽管可能是一种基于评分的相似度衡量方法，如余弦相似度。

吸引力的并且它有很高的历史价值。对残障人士来说更有吸引力和历史价值[6]。”在这个推荐系统中提供了结构化的概览,根据它们的适用性向不同的旅游者分类展示(如小孩、残障人士)。用户可以添加这些,并将这些结果加入他们的旅行日程,但是没有交互模型来修改以后的推荐结果。

据我们所知,没有其他系统用人口统计风格的解释。可能是因为人口信息的敏感性;有趣的是,我们可以想象很多用户不希望根据他们的性别、年龄或者种族来进行推荐(如“我们为您推荐电影‘Sex in the City’是因为您是一位 20~40 岁的女性”)。

## 15.8 总结与展望

在这一章,我们为推荐系统解释设计者提供了设计指南。首先,设计者应该考虑解释带来的好处,正是如此要评估解释的指标(如透明度、可读性、可信度、有效性、说服力和满意度)。开发人员可以选择彼此相关的多个标准,也可以选择相互互斥的多个标准。如果选择的标准互斥,区分评估标准就特别重要。在最近的研究中这种权衡体现了出来,且变得越来越明显[20, 61]。

此外,当评估解释时,系统设计者应该考虑他们要选择的指标,解释的依赖关系可能与系统各模块的依赖关系不同,如推荐被展示的方式(如最优物品、最优 N 物品、最优项相似物品、为所有物品预测并评分、结构概览),用户与解释交互的方式(例如,用户指定他们的需求,要求变更,给物品评分,提供他们的参考意见或者使用一个混合交互界面)和底层推荐引擎。

如果要提供一个简单的例子来说明解释与其他推荐系统因素之间的关系,我们就可以想象一个准确度较低的推荐引擎。当用户不喜欢最终推荐给他们的物品时,这可能会影响系统中有效性的所有计算。然而,这些计算不会反映解释本身的有效性。这种情况下,需要采用分层的评估方法[47],解释与推荐算法是相互独立的[61]。同样,应该考虑展示推荐结果的方法,交互的方法可能会影响解释的评估。

我们提供了一些例子讲解受常见算法影响的解释风格(如基于内容、协同、人口统计、基于知识/自然语言)以及他们在现有系统中的应用。从某种程度上来说,不同的解释风格可以混合使用(这样会牺牲透明度),并用于复杂的推荐方法,如潜在语义分析,但这些领域的研究有待展开。这些领域中初步可以做的一些工作可以在文献[33](解释马尔可夫决策过程)和[31](解释潜在语义分析)中找到。

然而,人们还没有对解释风格与它们在解释目的上的效果进行比较。这是一个研究方向,我们希望在不久的将来能有进一步的成果。同样,将来的工作可能会涉及更高级的界面。例如,树状结构(见图 15.9[4])为我们提供一个搜索空间的概览[9]。这种概览的风格也可用于解释。例如,一个用户被推荐了文章“The Votes Obama Truly Needs”,而该文章正是在这个概览中凸显出来的内容。界面“解释”这篇文章被推荐的原因是这个用户近期比较关注美国新闻(橘黄色),而它是热门的新闻(面积大),且是比较新的(明亮的颜色)。

最后,研究员开始发现解释是循环过程的一部分。解释会影响用户对推荐系统的想法,反过来又作用于解释的界面上。事实上,这也可能影响推荐的准确性[7, 20]。例如,文献[7]发现用户在新闻推荐系统里从自己的个人信息里删除关键字,会导致推荐精度下降。理解这种循环周期可能会成为将来研究的方向之一。

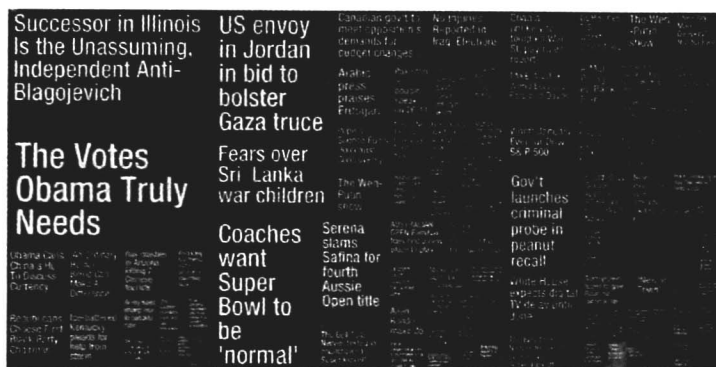


图 15.9 新闻地图——一个可视化新闻树。不同的颜色代表主要区域，面积和字体大小代表对当前用户的重要性，每个主题颜色的阴影代表新颖性

## 参考文献

1. Pandora (2006). <http://www.pandora.com>
2. Movielens dataset (2009). <http://www.grouplens.org/node/73>
3. Netflix dataset (2009). <http://www.netflixprize.com/>
4. Newsmap (2009). <http://www.marumushi.com/apps/newsmap/index.cfm>
5. Nutking (2010). <http://nutking.ectrldev.com/nutking/jsp/language.do?action=english>
6. Adrisson, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: Personalized recommendation of tourist attractions for desktop and handheld devices. *Applied Artificial Intelligence* **17**, 687–714 (2003)
7. Ahn, J.W., Brusilovsky, P., Grady, J., He, D., Syn, S.Y.: Open user profiles for adaptive news systems: help or harm? In: WWW '07: Proceedings of the 16th international conference on World Wide Web, pp. 11–20. ACM Press, New York, NY, USA (2007)
8. Andersen, S.K., Olesen, K.G., Jensen, F.V.: HUGIN—a shell for building Bayesian belief universes for expert systems. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990)
9. Bederson, B., Shneiderman, B., Wattenberg, M.: Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Transactions on Graphics* **21**(4), 833–854. (2002)
10. Bennett, S.W., Scott, A.C.: The Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project, chap. 19 - Specialized Explanations for Dosage Selection, pp. 363–370. Addison-Wesley Publishing Company (1985)
11. Bilgic, M., Mooney, R.J.: Explaining recommendations: Satisfaction vs. promotion. In: Proceedings of the Workshop Beyond Personalization, in conjunction with the International Conference on Intelligent User Interfaces, pp. 13–18 (2005)
12. Billsus, D., Pazzani, M.J.: A personal news agent that talks, learns, and explains. In: Proceedings of the Third International Conference on Autonomous Agents, pp. 268–275 (1999)
13. Burke, R.: Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* **12**(4), 331–370 (2002)
14. Burke, R.D., Hammond, K.J., Young, B.C.: Knowledge-based navigation of complex information spaces. In: AAAI/IAAI, Vol. 1, pp. 462–468 (1996)
15. Carenini, G., Mittal, V., Moore, J.: Generating patient-specific interactive natural language explanations. *Proc Annu Symp Comput Appl Med Care* pp. 5–9 (1994)
16. Chen, L., Pu, P.: Trust building in recommender agents. In: WPRSIUI in conjunction with Intelligent User Interfaces, pp. 93–100 (2002)
17. Chen, L., Pu, P.: Hybrid critiquing-based recommender systems. In: Intelligent User Interfaces, pp. 22–31 (2007)
18. Cosley, D., Lam, S.K., Albert, I., Konstan, J.A., Riedl, J.: Is seeing believing?: how recommender system interfaces affect users' opinions. In: CHI, *Recommender systems and social computing*, vol. 1, pp. 585–592 (2003).
19. Cramer, H., Evers, V., Someren, M.V., Ramlal, S., Rutledge, L., Stash, N., Aroyo, L., Wieling, B.: The effects of transparency on perceived and actual competence of a content-based recommender. In: Semantic Web User Interaction Workshop, CHI (2008)

20. Cramer, H.S.M., Evers, V., Ramlal, S., van Someren, M., Rutledge, L., Stash, N., Aroyo, L., Wielinga, B.J.: The effects of transparency on trust in and acceptance of a content-based art recommender. *User Model. User-Adapt. Interact* **18**(5), 455–496 (2008).
21. Czarkowski, M.: A scrutable adaptive hypertext. Ph.D. thesis, University of Sydney (2006)
22. Doyle, D., Tsymbal, A., Cunningham, P.: A review of explanation and explanation in case-based reasoning. Tech. rep., Department of Computer Science, Trinity College, Dublin (2003)
23. Felfernig, A., Gula, B.: Consumer behavior in the interaction with knowledge-based recommender applications. In: *ECAI 2006 Workshop on Recommender Systems*, pp. 37–41 (2006)
24. Fogg, B., Marshall, J., Kameda, T., Solomon, J., Rangnekar, A., Boyd, J., Brown, B.: Web credibility research: A method for online experiments and early study results. In: *CHI 2001*, pp. 295–296 (2001)
25. Fogg, B.J., Soohoo, C., Danielson, D.R., Marable, L., Stanford, J., Tauber, E.R.: How do users evaluate the credibility of web sites?: a study with over 2,500 participants. In: *Proceedings of DUX'03: Designing for User Experiences*, no. 15 in *Focusing on user-to-product relationships*, pp. 1–15 (2003). URL <http://doi.acm.org/10.1145/997078.997097>
26. Ginty, L.M., Smyth, B.: Comparison-based recommendation. *Lecture Notes in Computer Science* **2416**, 731–737 (2002).
27. Hance, E., Buchanan, B.: Rule-based expert systems: the MYCIN experiments of the Stanford Heuristic Programming Project. Addison-Wesley (1984)
28. Häubl, G., Trifts, V.: Consumer decision making in online shopping environments: The effects of interactive decision aids. *Marketing Science* **19**, 4–21 (2000)
29. Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. In: *ACM conference on Computer supported cooperative work*, pp. 241–250 (2000)
30. Hingston, M.: User friendly recommender systems. Master's thesis, Sydney University (2006)
31. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: *ICDM (2008)*
32. Hunt, J.E., Price, C.J.: Explaining qualitative diagnosis. *Engineering Applications of Artificial Intelligence* **1**(3), Pages 161–169 (1988)
33. Khan, O.Z., Poupart, P., Black, J.P.: Minimal sufficient explanations for mdps. In: *Workshop on Explanation-Aware Computing associated with IJCAI (2009)*
34. Krulwich, B.: The infofinder agent: Learning user interests through heuristic phrase extraction. *IEEE Intelligent Systems* **12**, 22–27 (1997)
35. Lacave, C., Diéz, F.J.: A review of explanation methods for bayesian networks. *The Knowledge Engineering Review* **17**:2, 107–127 (2002)
36. Lacave, C., Diéz, F.J.: A review of explanation methods for heuristic expert systems. *The Knowledge Engineering Review* **17**:2, 107–127 (2004)
37. Lewis, C., Rieman, J.: Task-centered user interface design: a practical introduction. University of Colorado (1994)
38. Lopez-Suarez, A., Kamel, M.: Dykor: a method for generating the content of explanations in knowledge systems. *Knowledge-based Systems* **7**(3), 177–188 (1994)
39. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: Thinking positively - explanatory feedback for conversational recommender systems. In: *Proceedings of the European Conference on Case-Based Reasoning (ECCBR-04) Explanation Workshop*, pp. 115–124 (2004)
40. McNee, S., Lam S.K. and Guetzlaff, C., Konstan J.A. and Riedl, J.: Confidence displays and training in recommender systems. In: *INTERACT IFIP TC13 International Conference on Human-Computer Interaction*, pp. 176–183 (2003)
41. McNee, S.M., Lam, S.K., Konstan, J.A., Riedl, J.: Interfaces for eliciting new user preferences in recommender systems. *User Modeling* pp. pp. 178–187 (2003)
42. McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: How accuracy metrics have hurt recommender systems. In: *Extended Abstracts of the 2006 ACM Conference on Human Factors in Computing Systems (CHI 2006)* (2006)
43. McSherry, D.: Explanation in recommender systems. *Artificial Intelligence Review* **24**(2), 179 – 197 (2005)
44. Nielsen, J., Molich, R.: Heuristic evaluation of user interfaces. In: *ACM CHI'90*, pp. 249–256 (1990)
45. Ohanian, R.: Construction and validation of a scale to measure celebrity endorsers' perceived expertise, trustworthiness, and attractiveness. *Journal of Advertising* **19**:3, 39–52 (1990)
46. O'Sullivan, D., Smyth, B., Wilson, D.C., McDonald, K., Smeaton, A.: Improving the quality of the personalized electronic program guide. *User Modeling and User-Adapted Interaction* **14**, pp. 5–36 (2004)
47. Paramythi, A., Totter, A., Stephanidis, C.: A modular approach to the evaluation of adaptive



- user interfaces. In: S. Weibelzahl, D.N. Chin, G. Weber (eds.) *Evaluation of Adaptive Systems in conjunction with UM'01*, pp. 9–24 (2001)
48. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review* **13**, 393–408 (1999)
  49. Pu, P., Chen, L.: Trust building with explanation interfaces. In: *IUI'06, Recommendations I*, pp. 93–100 (2006).
  50. Pu, P., Chen, L.: Trust-inspiring explanation interfaces for recommender systems. *Knowledge-based Systems* **20**, 542–556 (2007)
  51. Rafter, R., Smyth, B.: Conversational collaborative recommendation - an experimental analysis. *Artif. Intell. Rev* **24**(3-4), 301–318 (2005). URL <http://dx.doi.org/10.1007/s10462-005-9004-8>
  52. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing. In: P. Funk, P.A. González-Calero (eds.) *ECCBR, Lecture Notes in Computer Science*, vol. 3155, pp. 763–777. Springer (2004)
  53. Roth-Berghofer, T., Schulz, S., Leake, D.B., Bahls, D.: Workshop on explanation-aware computing. In: *ECAI* (2008)
  54. Roth-Berghofer, T., Tintarev, N., Leake, D.B.: Workshop on explanation-aware computing. In: *IJCAI* (2009)
  55. Sinha, R., Swearingen, K.: The role of transparency in recommender systems. In: *Conference on Human Factors in Computing Systems*, pp. 830–831 (2002)
  56. Sørmo, F., Cassens, J., Aamodt, A.: Explanation in case-based reasoning perspectives and goals. *Artificial Intelligence Review* **24**(2), 109 – 143 (2005)
  57. Swearingen, K., Sinha, R.: Interaction design for recommender systems. In: *Designing Interactive Systems*, pp. 25–28 (2002).
  58. Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: Justified recommendations based on content and rating data. In: *WebKDD Workshop on Web Mining and Web Usage Analysis* (2008)
  59. Thompson, C.A., Göker, M.H., Langley, P.: A personalized system for conversational recommendations. *J. Artif. Intell. Res. (JAIR)* **21**, 393–428 (2004).
  60. Tintarev, N., Masthoff, J.: Over- and underestimation in different product domains. In: *Workshop on Recommender Systems associated with ECAI* (2008)
  61. Tintarev, N., Masthoff, J.: Personalizing movie explanations using commercial meta-data. In: *Adaptive Hypermedia* (2008)
  62. Vig, J., Sen, S., Riedl, J.: Tagsplanations: Explaining recommendations using tags. In: *Intelligent User Interfaces* (2009)
  63. Wang, W., Benbasat, I.: Recommendation agents for electronic commerce: Effects of explanation facilities on trusting beliefs. *Journal of Management Information Systems* **23**, 217–246 (2007)
  64. Wärnestål, P.: Modeling a dialogue strategy for personalized movie recommendations. In: *Beyond Personalization Workshop*, pp. 77–82 (2005)
  65. Wärnestål, P.: User evaluation of a conversational recommender system. In: *Proceedings of the 4th Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, pp. 32–39 (2005)
  66. Wick, M.R., Thompson, W.B.: Reconstructive expert system explanation. *Artif. Intell.* **54**(1-2), 33–70 (1992).
  67. Ye, L., Johnson, P., Ye, L.R., Johnson, P.E.: The impact of explanation facilities on user acceptance of expert systems advice. *MIS Quarterly* **19**(2), 157–172 (1995).
  68. Yee, K.P., Swearingen, K., Li, K., Hearst, M.: Faceted metadata for image search and browsing. In: *ACM Conference on Computer-Human Interaction* (2003)
  69. Zaslow, J.: Oh no! My TiVo thinks I'm gay (2002).

# 基于实例评价研究的产品推荐系统的可用性准则

Pearl Pu、Boi Faltings、Li Chen、Jiyong Zhang 和 Paolo Viappiani

**摘要** 在过去的十年中，我们团队开发了一套基于实例评价的决策工具，来帮助用户在电子购物中发现自己更喜欢的产品。在本章中，我们将调研实例评价相关的可用性研究工作，并衍生出一套可用性指导准则，来总结这方面的主要结果。我们的调研侧重于用户与系统的三个关键交互过程：初始偏好提取过程、偏好修正过程、推荐系统结果的展示过程。为了对这些准则的推导提供基础，我们根据三个交互参数：准确性、信任度、代价 (Accuracy Confidence Effort, ACE)，开发了一个多目标框架。我们采用这个框架分析了过去的工作，并对每个准则给出了特定的应用环境：什么时候系统应最大化地增加用户的决策精度，什么时候提高用户信任度，什么时候应最小化与用户交互的代价。由于这一多准则模型的通用特性，我们提出的这些准则，将有助于其他推荐系统，特别是电商环境中的可用性设计过程。我们在这一领域首创的 ACE 框架，同样也适用于评估基于用户兴趣推荐系统的效果。这里提出的 ACE 框架也是这一领域首次从用户中心的视角评估基于偏好推荐系统效果的方法。

因此，开发者可以通过这些准则，开发出效果更好、更成功的产品推荐系统。

## 16.1 简介

著名的可用性研究者 Jacob Nielsen 曾提出：电商的第一定律是，如果用户不能发现一个产品，那么他们就不可能会买它<sup>①</sup>。这里的“发现”实际上是为电子商务系统所必须满足的挑战性任务做了一个定义。它指的是在线零售商帮助用户找到满意产品 (有时甚至自己不知道)，并鼓励用户购买推荐产品的能力。也就是说，系统不仅需要帮助用户寻找，而且需要辅助其做出购买决定。

当用户在网上环境中碰到上述问题时，他们究竟会怎么做呢？随着愈演愈烈的竞争，在线零售商必须维护一个逐渐增多的有效产品集。新的物品会定期添加到库存目录中，来保证直接竞争者的产品也在他们自己的库存中。查找想要的选择这一任务直接依赖于可供选项的数量。事实上随着无限货物仓库这一概念的推广，对一般用户而言，这一任务即使

---

Pearl Pu · Li Chen · Jiyong Zhang Human Computer Interaction Group, School of Computer and Communication Sciences, Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne, Switzerland e-mail: \{pearl.pu, li.chen, jiyong.zhang\}@epfl.ch

Boi Faltings, Artificial Intelligence Laboratory, School of Computer and Communication Sciences Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne, Switzerland e-mail: boi.faltings@epfl.ch

Paolo Viappiani, Department of Computer Science, University of Toronto, 6 King's College Road, M5S3G4, Toronto, ON, CANADA e-mail: paolo.viappiani@gmail.com

翻译：何伟岗，胡聪(胡户主) 审核：张洋，左其盛

① Nielsen 在 2003 年 Alertbox 中陈述了这一规则，详情请参考 <http://www.useit.com/alertbox/20030825.html>。

不是不可能,也将变得更加艰巨。在这种情况下,用户倾向于使用以下两种决策方法中之一。第一种情况,他们试图得到更高的决策准确率,但却需要筛选所有选项,并在产品各个方面优缺点之间权衡上耗费大量时间。另一方面,他们也可以采用启发式的策略,更有选择性地处理信息。这种情况下,尽管他们花费了更少的工作,这些启发式策略也可能导致决策失误甚至后悔。总而言之,在准确率和代价平衡关系上,没有哪种方法是理想最优的。参考[40],如何在决策中把握好准确率和代价的平衡,是个不好调解的工作。

有些知名的研究工作,提出了如何为用户开发交互高效和智能的工具。因此,基于偏好的推荐系统出现了,其有效的搜索和导航机制,在电子商务及其他决策环境中,得到广泛认可。在过去的十年中,我们提出了实例评价的方法,开发了一套基于该方法的决策工具,为产品搜索问题[43, 45, 51, 52]提供个性化和推荐。目前已经开展并发表了十多项用户调查研究,在出行计划、公寓搜索、数码产品的检索等领域验证了该方法。但是,在电子商务领域得到广泛应用的实例评价方法仍然存在一定的局限性。

我们的目标是通过分析和调查我们之前实例评价方向上的工作,通过总结出一套可用性准则,综合了主要成果。更特别的是,我们主要关注初始偏好提取过程、偏好修正过程、推荐系统结果的展示过程这三个关键交互过程。为了对准则的导出提供基准,我们调研了一个产品推荐系统为了最大化满足用户需求以及使用推荐系统意愿所必须满足的目标。推荐系统的准确率,也就是系统如何找到用户真正想要的物品,通常是推荐系统的一个重要的中心内容。但是,准确率并没有完全考虑用户的利益,因为准确率的提高有可能是通过牺牲系统的可用性达到的[35]。我们都知道,当人们认知能力有限时,用户很难得到高精度的准确率,因为这需要的代价是更高的。由于产品检索是个决策过程,推荐系统必须让用户相信系统推荐给他的一定是自己真正想要的(信任度)。因此,不仅包括准确率,我们提出了称为 ACE 的多目标框架,得到了可用性准则:1)系统帮助用户发现最喜欢物品的能力(Accuracy, 准确率);2)系统激发用户有信心选择推荐给他们的物品的能力(Confidence, 信任度);3)实现相应准确率用户所需的总代价(Effort, 用户代价)。

考虑这样一种情况,部分用户愿意花费大量的时间来获得非常准确的推荐结果,而部分用户可能愿意接受低准确率但是快速的结果,因此,推荐系统的设计者必须权衡好准确率和代价,这也是权衡信任度和代价的过程,因为用户如果更相信推荐结果,则其可能更愿意接受更多的交互代价。可见,推荐系统设计的最大挑战是如何让用户坚信,只要他们花费足够的代价,就能获得可接受的推荐结果。

得到这些准则的这三个条件同样也可以作为评估框架来衡量一个产品推荐系统的可用性。因此本章在推荐系统的两个主要方面作出了贡献。从学术角度看,本章提出了一套以用户为中心的标准,来评估基于偏好推荐系统的效果。从实践角度看,本章在三种关键交互行为上调研了实例评价的现状,得出 11 个更具推广型和可扩展性的可用性准则。由于这些准则源自的方法已经在可用性研究中得到了验证,所以实际开发者可以在产品推荐系统(包括设计阶段和测试阶段)的可用性设计过程中大胆采用。

## 16.2 预备知识

### 16.2.1 交互模型

基于用户偏好的推荐系统,是根据用户对于物品属性偏好的显式表述来向用户推荐物品的。虽然本章描述的推荐系统是使用不同的体系结构以及使用不同的数据集来进行评估

的,但是,特别是在交互模型上,它们都有一些共同的特征。为了梳理贯穿整个调研所用到的术语和准则推导过程,我们在图 16.1 中给出了能概括推荐系统各个阶段的交互通用模型。用户会通过像图形用户界面这样的形式来设置一组初始偏好开始交互过程。然后,推荐系统根据采集到的用户信息,过滤掉无效选择,然后根据用户偏好,给用户返回推荐的结果(Recommendation Set, 推荐集合)。接着,有两种情况,要么用户从推荐集合中找到最喜欢的物品,结束与系统的交互,要么用户修改偏好模型以获得更精确的推荐,如使用类似“我想要更便宜的物品”的评价。最后的用户反馈步骤称为偏好修正。最终,用户或者选择了单个物品,或者构建了一个可用于进一步比较的考虑集合。

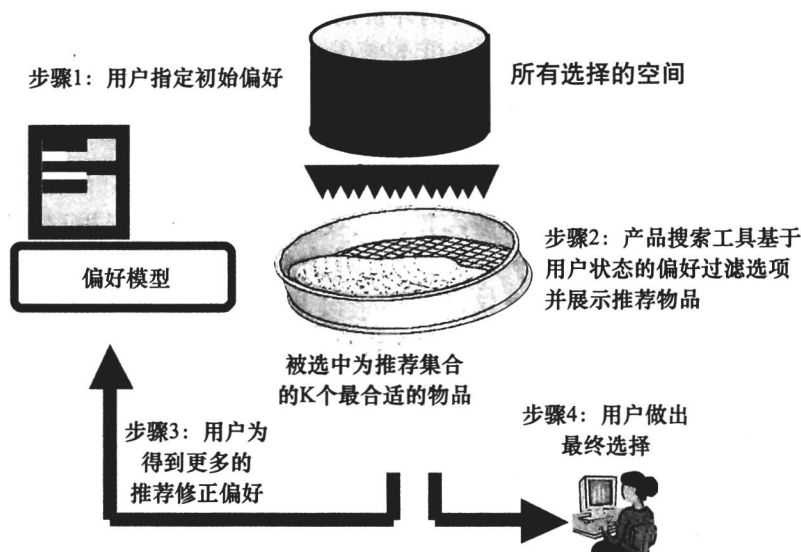


图 16.1 典型的用户系统交互过程

当然,如果用户对这一过程确实能找到满意的产品不抱有足够的信心,而且因此不愿完成充分的偏好修正循环让系统产生精确的偏好模型,这样推荐系统就可能失败了。我们的指导准则可以避免这种情况,本章最后将提供一个模型,根据用户的决策过程,对该模型进行基本阐释。

推荐集合生成后,系统可以采取多样的策略来展示这些结果。一个典型的展示方法就是,在  $m$  个交互过程中,每个过程展示  $k$  ( $1 \leq k \leq n$ ,  $n$  是产品总数) 个物品。在每次展现这  $k$  个物品时,用户都会挑出她的目标结果加入考虑集合中。当然,过程越多,用户需要花费的代价也就越多。另一方面,如果展示的集合较少,用户可以整体浏览一遍来得到目标结果,进而参与更多的交互过程。这个代价和准确率的权衡问题,在个性化推荐系统中会被详细讨论到。

上面提到的各种交互过程可能不会同时出现在相同的系统里。例如,初始偏好提取过程可以是一个可选的步骤,用户一访问站点就被提供一组推荐结果(如各个类目下的畅销品)。另一方面,其他系统可能会提取用户的初始偏好,但不提供选项允许用户修改它们。

## 16.2.2 基于效用的推荐系统

由于实例评价工具是基于多重属性效用理论的,我们将给出这一基础推荐算法的综述。

这类推荐系统的潜在的基本假设是,人们因为物品的某些属性从而对该物品有偏好。一个属性的不同数值对应着不同的偏好程度,如当客人很多时,大型的公寓会很有用,可

是当需要打扫时,这就成为了缺点。因此,用户需要根据物品每项特点有用或无用的程度来权衡其优点和缺点,即偏好是属性的加权函数。

形式上,基于偏好的推荐问题可以描述为多重属性决策问题(Multi-Attribute Decision Problem, MADP):  $\Psi = \langle X, D, O, P \rangle$ , 其中  $X = \{X_1, \dots, X_n\}$  指某类产品的属性集合,  $D = D_1 \times \dots \times D_n$  指产品在每项属性上的值域( $X_i$  对应的值域为  $D_i (1 \leq i \leq n)$ ),  $O = \{O_1, \dots, O_m\}$ , 指当前库存物品集合,  $P = \{P_1, \dots, P_i\}$  为用户的偏好集。MADP 的目标是找到一种用户最感兴趣的产品, 该类问题可以使用基于约束方法和基于效用的方法来处理。接下来我们将引出基于多属性效用理论(MAUT)的方法来解决一个给定的 MADP 问题, 基于限制的方法请参考文献[45]。

### 16.2.2.1 多属性效用理论(MAUT)

效用理论要追溯到 1738 年, 当时 Bernoulli 从货币价值效用的角度提出了他对于 St. Petersburg 悖论的阐释。两个世纪之后, von Neumann 和 Morgenstern 重新使用此方法来解决他们在经济学中遇到的问题[71]。后来, 20 世纪 50 年代, Marschak[29]、Herstein 和 Milnor[21], 以形成 Von Neumann Morgenstern 定理(VNM 定理)[39, 59]根基的一套公理为基础, 建立了期望效用理论。

20 世纪 70 年代, Kneeney 和 Raiffa[24]扩展了效用理论, 能够支持多属性的情况。多属性效用理论的主要思想是将用户偏好表示为属性值的效用函数。

使用符号  $\geq$  表示用户偏好顺序, 如  $A \geq B$  表示用户对  $A$  的偏好程度高于  $B$ 。根据效用理论, 对于一个给定的 MADP 问题, 存在一个效用函数  $U: O \rightarrow \mathfrak{R}$ , 对于任意产品集合  $O$  中的产品  $O$  和  $\bar{O} \in O$ , 存在以下等价关系:

$$O \geq \bar{O} \Leftrightarrow U(O) \geq U(\bar{O}) \quad (16.1)$$

并且, 产品  $O$  表示为一系列属性的集合  $\langle X_1 = x_1, \dots, X_n = x_n \rangle$  (简写为  $\langle x_1, \dots, x_n \rangle$ ), 因此上述公式等价于:

$$\langle x_1, \dots, x_n \rangle \geq \langle \bar{x}_1, \dots, \bar{x}_n \rangle \Leftrightarrow U(\langle x_1, \dots, x_n \rangle) \geq U(\langle \bar{x}_1, \dots, \bar{x}_n \rangle) \quad (16.2)$$

可见, 如果给出了效用函数, 那么根据上述公式, 任意的产品的偏好度都可以用数值效用表示, 也能对所有产品根据偏好排序。

因此, 找出能够准确表示用户偏好的效用函数  $U$ , 是一项很有挑战性的工作。虽然理论上效用函数可以用任何形式表示用户偏好, 但是经常使用其中一种来减少计算代价。如果各个偏好都是相互独立的<sup>①</sup>, 那么效用函数具有如下的附加形式:

$$U(\langle x_1, \dots, x_n \rangle) = \sum_{i=1}^n w_i v_i(x_i) \quad (16.3)$$

其中,  $v_i$  是属性  $X_i$  上的值函数(范围是  $[0, 1]$ ),  $w_i$  是属性  $X_i$  的权重,  $\sum_{i=1}^n w_i = 1$ , 也就是说产品  $O$  的效用函数是所有属性值的权重的总和。每项属性的权重, 缺省值可以是  $1/n$ , 可以允许用户设置多项属性上的权重。值函数  $v_i$  则由用户对属性  $X_i$  的偏好决定。通常具有形式为  $v_i = ax_i + b$  的线性函数足以用来表示用户在每一个属性上的偏好。

一旦每项物品的效用函数都确定了, 我们可以根据效用值对产品进行排序, 并选出前  $K$  项最高的物品作为推荐集合。在实践中, 我们假定任一产品的各项属性都是独立的, 因

① 如果属性  $X$  的值不依赖于属性  $Y$ , 则称  $X$  独立于  $Y$ , 如果属性  $Y$  的值也不依赖于  $X$ , 则称  $X$  与  $Y$  相互独立, 文献[22]有更多具体细节。



此也可以使用上述的效用函数的附加形式。

### 16.2.3 准确率、信任度和代价的框架

在引言中提到,我们关于多目标需求框架的研究,是为了探寻如何演化设计指导原则的基本原理。我们寻找产品推荐系统必须满足的标准,来实现最大满足用户需求和最大提高用户使用系统的意愿。因此,我们不仅给出了 ACE(Accuracy、Confidence、Effort)框架更精确的定义,也给出了衡量这三个变量的方法。

准确率是指推荐系统的客观准确率。对于基于评分的系统(Rating-based Systems),最常用的衡量方法是平均绝对偏差(Mean Absolute Error, MAE)[1]。它是通过对数据集的离线的计算步骤来衡量的,也就是俗称“留一法”(leave-one-out)。“留一法”通过先去除一个评分,用被评估的推荐系统算法预测这一评分。这一估算评分将会与实际评分进行对比,并计算两者的绝对偏差值。重复将这一步骤用于所有的评分,然后计算出的平均值称为平均绝对偏差。

基于效用的推荐系统产生的推荐结果,是基于用户偏好数据的,而这些数据是不可能被模拟出的,因此,离线的衡量准确率的方法也是行不通的。这一领域,常用的一个方法是文献[48]定义的任务更换(switching task)。它通常可以估量出用户偏好物品有多少是出自系统的推荐结果。例如,如果在实验操作者给出所有的可能选项之后,有70%的用户没有改变他们的决定,认为在推荐过程中发现了他们偏好的物品,则系统的推荐率是70%。因此,实验第一步,每个用户需要与推荐系统交互以选出考虑集合。第二步,实验操作者需要展示所有存在的物品,然后用户判定考虑集合中所有物品是否仍有吸引力。如果他更换了其他的物品,则系统未能帮助他做出准确的决策。类似于这样的过程,在消费者决策中被用于评判决策质量,称为任务更换[20]。更换物品的用户比例,称为更换率,它是唯一能够计算出个性化推荐工具的决策准确率的精度的方法。可是,由于这个方法是很耗时的,任务更换方法仅用于精心定义的实验研究中。

用户信任度是 ACE 框架的第二个评估准则,说明了系统诱导用户选择推荐结果的能力。这是一个主观变量,只能通过研究后期的调查问卷得到,让用户给出满意度描述,如我相信系统推荐给我的正是我想要的。由于客观准确率很难得到,用户的信任度常常用于评估系统的感知准确率。感知准确率,与实际的准确率有很强的相关性,但同样也会被系统的其他因素所影响:用户是否被给予了足够的选择并被充分地包含在偏好诱导和修正过程中,推荐结果的展现形式是否让用户确信已经发现最好的结果。

用户代价包含了两个时间花费,第一个是用于完成引导行为,如偏好提取和偏好修正的所需时间,第二个是用户构造考虑集合所需时间。我们也可以为任意交互行为衡量交互周期的次数来替换花费的时间,这样度量就与单个用户的活动习惯就无关了。

本章分析了基于实例推荐系统三个主要步骤:偏好提取、偏好修正和结果展示,然后使用准确率、信任度和代价来评估不同的推荐技术,然后衍生出 11 条成功的基于实例推荐系统中的指导准则,最后与提供统一动机的用户行为模型进行对比。

我们相信,用户信任度是评判任何推荐系统效果的主要因素,因此我们认为 ACE 框架同样适用于评判其他类型的推荐系统。

### 16.2.4 本章结构

我们根据图 16.1 描述的三个过程:偏好提取(第一步)、偏好修正(第四步)、展示策

略(第二步),对上述 11 条指导原则进行分类。接下来的章节包括:16.3 节回顾了其他领域中,偏好提取过程和个性化推荐工具的设计准则。16.4 节介绍了激发用户尽可能准确地描述其初始偏好的准则。16.5 节继续讨论偏好提取和标识出具体方法,来帮助用户描述出其完整的喜好。16.6 节阐述了帮助用户解决偏好冲突和决策权衡的策略。16.7 节提出了展示策略的一些准则,以便能够实现较低的用户交互代价,较高的决策精度和信任度。16.8 节提出的用户行为模型对这些准则给出了理论依据。16.9 节总结全文。

## 16.3 相关工作

### 16.3.1 推荐系统分类

有两类推荐系统已经广为人知,并且广泛用于收集和构建用户偏好数据集。一类是基于用户显性描述偏好的推荐系统(Preference-based Recommender),另一类是基于用户导航和购买行为的推荐系统(Behavior-based Recommender)。基于行为的推荐系统生成推荐结果时会根据用户累积的交互行为,如用户浏览和购买过的物品,请参考文献[75]对它的论述,并参考[26, 55]对基于人口统计的推荐系统的论述。基于偏好的推荐系统又包含四种类型:基于评分的,基于案例的,基于效用的和基于评价的。更多推荐系统的分类方法请参考[1, 7]。

### 16.3.2 基于评分的推荐系统

用户通过对他们接触过的物品给出二元或多尺度分数的方式,显式地表达他们的偏好(甚至他们自己都不知道)。要么是系统向用户提供一组物品评分,要么是用户自己选择一组物品评分。这些初始评分就构成了用户资料。由于用户是根据相似兴趣或历史偏好被推荐的,这类系统又通常称为协同过滤推荐系统。因为这个原因,这类系统也被称为社会化推荐系统。协同算法的工作的细节可以在文献[1]中找到。最近有些网站,如 tripadvisor.com,通过使用同一物品多项属性的评分来得到更精确的用户偏好资料。

### 16.3.3 基于案例的推荐系统

这类系统将用户感兴趣物品的相似物品推荐给用户。每个产品都可看作拥有多重属性的案例。基于内容[1]和基于案例[7]的技术被用于分析有效产品的属性值和用户已知偏好,并根据排序方法确定一个或多个最优排序选项。

### 16.3.4 基于效用的推荐系统

基于效用的推荐,如基于实例评价的推荐系统,是基于用户对多重属性产品的偏好描述推荐物品。多属性产品适用于将所有有效数据都用相同的属性集合 $\{a_1, \dots, a_k\}$ 来表示的编码方案,其中每个属性 $a_i$ 可以是相应 $d(a_i)$ 值集中任何一个值 $v$ 。如电子市场上所有的数码相机可以被分为以下属性:制造商、价格、分辨率、镜头、内存、屏幕大小、厚度和质量等。可见,不同的商品类型拥有不同的属性集合。我们假定用户的偏好是完全依赖于商品的属性值,因此两个属性值相同的商品,它们对于用户拥有相同的偏好度。另外,这里提到的商品,如数码相机、笔记本和公寓,都会要求有一定的财务负担。这些商品称作高参与度商品,因为用户被期望具有足够的诚意与系统交互,参与选择过程,并花费一

定数量的代价处理信息[62]。并且在这种环境下,相比于书籍、DVD、文章等更简单的物品,用户可能会有更复杂的决策行为。

使用这些技术的推荐系统也称作基于知识的推荐系统[7]和基于效用的决策支持系统(Decision Support Interface Systems, DSIS)[62]。这里的效用,是指那些被用来计算产品是否满足用户偏好的多重属性效用理论。一个相关的技术,特别是用于检索可配置产品上,使用了约束补偿的技术[45]。基于效用的(UBR)和基于案例的(CBR)推荐系统,其区别在于效用这两个字,对于前者,用户偏好更重要,但却被后者忽视。此外,重要性权衡的概念是 UBR 决策的重要部分。本质上 UBR 推荐的是决策,而不仅仅是相似产品。详细信息请看本节中的权衡推理。

### 16.3.5 基于评价的推荐系统

基于案例的和基于效用的推荐系统,都可以通过增加额外的评价的交互步骤,来提升推荐效果。基于评价的产品推荐系统,通过模拟一个虚构的卖家,来根据用户当前偏好推荐可选集,然后根据类似于“我想要更便宜的”或“更快的处理速度”的评价形式,来收集用户的反馈信息。这些评价方法,可以通过预测在下一次推荐周期中的需求,来帮助代理者提高准确率。可能需要多次周期,直到用户最终找到了理想商品。用户不太可能描述出他们所有的偏好,特别是对于不熟悉的商品,因此偏好评价代理是个有效的方法,从而能够帮助用户逐步构建偏好模型,并且随着用户有更多的候选项进而不断地得到修正。

### 16.3.6 其他设计指导准则

基于我们团队最近在进行基于偏好的检索和推荐系统的交互技术的研究,本章推导出了一套可用性设计准则。在本节中,我们将不再回顾对积累这些准则列表有过贡献的相关工作,而是将这些讨论放到本章各处有关如何构造这些准则的地方。我们将描述两篇论文,这两篇论文在决策支持系统方面,提出如何推导出良好的设计准则。第一项参考的工作,从市场调查角度出发,抽象出了一套推荐工具,能够提高用户参与在线或网站系统交互的积极性。第二项参考的工作,在设计高风险结果决策问题上,描述了一系列“构建代码”的指导准则,来提升偏好构建的效果。

根据对那时可用的基于偏好的产品搜索和推荐工具虽显挑剔但也合情合理的看法,Spiekermann 和 Paraschiv 提出了 9 项推荐设计建议用来模拟和增强用户与系统之间的交互意愿[62],但是,这些建议与我们的准则没有重叠的地方。

这些建议中的大部分标准,都是从与信息检索相关的营销文献和感知风险理论得来的,该理论将用户与产品推荐系统交互的意愿定义为她想降低与购买决策相关的各种机能上、财务上和情感上风险的动机。因此,这些建议是从减轻用户各项风险角度抽象出来的。与我们的工作相比,这是一个水平的方法论,它覆盖了整个电子商务网站的设计过程,而推荐系统这一手段只是其中的主要技术组件。而我们对推荐系统引擎的交互技术进行深入的探究,是一个垂直的方法论,保证消费者能够在与推荐系统交互时的偏好提取和决策中,得到最理想的可用的支持。

虽然做出高风险决策是更应该受到关注的主题,Payne 等[41]提出的评价方法,在解决消费者面临购买决策问题的需求和偏好问题上和我们的目标是类似的。他们的工作挑战了在精细化和预先偏好上的许多传统假设,也提出了构建偏好的理论。在新的框架下,作

者提出了一套“构建代码”(类似于指导准则),来帮助专业的决策者建立价值函数和做出高质量的决策。当然,在基于人类偏好的天性上,构建和评估偏好方法上,以及如何处理好平衡来得到合理的决策,他们的相关研究,深深地影响了我们的工作。本章的相关部分将详细介绍行为决策研究中这部分以及其他工作的参考文献。

## 16.4 初始偏好提取

实例评价系统是一个决策工具,可以帮助用户寻找到多属性的商品,如航班、数码相机、笔记本电脑等。我们使用  $P = \{(a_i, w_i)\} (1 \leq i \leq n)$  来描述基于效用的推荐系统中,用户对于整个  $n$  维属性集商品的偏好, $a_i$  表示第  $i$  维属性的要求指标, $w_i$  表示该项指标的满足程度。这个模型在文献[24]中也被称为价值函数。而很多推荐系统称为用户偏好模型,当然,有些方法假定所有的属性值拥有相同的权重,因此不会抽取这些信息[9, 53]。偏好提取(也被称为查询指定),是获得用户信息的初始化过程。

很明显,用户的偏好也可以通过让用户回答一些简单的问题来获得。许多在线的检索工具使用图形化界面填表或自然语言对话框的方式来收集用户的这些信息。这些情况下,用户被要求陈述他们在各个方面的偏好,如起飞和达到时间、航线和中转机场等,并且必须填写所有的选项,这种方法也被称作非递进式的,因为所有偏好必须在开始时被收集到。

这种简单的方法为什么会不管用呢?先让我们从行为决策的角度来试着理解用户的偏好表达方式。根据自适应决策理论[40],用户的偏好也是自适应的,也是与当前的决策现状息息相关的。根据这一原则,用户在没有得到任何利益前[62],可能会不太愿意去回答那些初始引导问题,也有可能他们缺乏足够的领域知识来正确回答问题。也就是说,如果系统在开始时,要求很重的选择过程,采集到的偏好信息有可能是不准的或是错误的。

相似的研究揭示,用户的偏好是依赖于上下文环境的,也是根据他们的需求,在逐步获得更丰富的领域信息后,逐步建立起来的[40, 41]。例如,Tversky 等做了一个关于要求受试者购买微波炉的用户调研[66],实验的参与者被分为两组,每组 60 人。第一组,每个用户都有两个选择,一个是 Emerson 售价 110 美元,另一个是 Panasonic 售价 180 美元,并且两者在标价上都显示当前价格是原价的  $2/3$ 。结果表明,仅有 43% 的参与者选择 180 美元的微波炉。第二组,除了上述两个选择外,还增加了一个 Pannasonic 当前售价 200 美元,是原价的 9 折。这种情况下,60% 的参与者选择了 180 美元的微波炉。很明显,仅仅是参考选项的改变,就促使了更多的人选择同样的物品。这一发现恰好说明了,用户的偏好在初始时是很难揭露出来的,但可以在逐步获得更丰富的信息后逐步建立起来。

为了验证这些典型的行为理论,是否在在线环境中依然有效,我们也进行了几项实验研究。在此,22 位参与者被要求与偏好提取界面进行交互[67],用户开始时只能描述平均 2.1 项属性(总共 10 项)上的自身的偏好情况,直到最后做出选择时,他们对自身偏好的各项属性值描述程度,也只达到了平均 4.19 项。

还有一项研究,也证明了用户不太可能在初始时描述出其自身所有的偏好。这项研究在用户交互过程中,对比了不递增选项和每次交互递增一个选项两种不同的情况下[68],用户检索商品的准确率和代价。这里,不递增选项的过程,是指用户需要在单一的图形界面上填写出其所有的偏好。而每次交互增加一个,是指每一个偏好结果都是由用户决定的。实验中,40 位参与者被随机均分为两组,每组被指派到一个需评估的系统(不递增或递增)中。非递增的方法中,用户可以描述出平均 7.5 项属性值(总共 10 项)上的偏好,而



递增的方式也是同样的结果。可是,同样的用户代价下,递增的方式拥有 70% 的准确率,而非递增只有 25%。也就是说,在非递增填表方式中,只有 25% 的用户找到了他们想要的商品。可见,非递增的方式可能产生出同样的数据,但是不一定能有同样的质量,当然,当用户的偏好仍然不确定时,用户可能不会提供准确的和持续的属性描述。

在基于协同过滤系统中的偏好提取过程,也有同样的发现。McNee 等对比了新用户对于电影评分三种不同的界面策略的效果[34]。第一种策略,系统让用户对选择的电影进行评分,来最大限度获得用户偏好模型,这里,系统决定了用户需要对哪些电影进行评分。第二种策略,用户可以对任意的电影进行评分。第三种策略,混合了前两种,用户可以选择前两种之一。总共 225 位新用户参与这项实验,结果说明,相比于其他两种策略,用户控制的策略(第三种)可以得到更高的推荐准确率。另外,用户控制的界面,更能促使用户再次回到系统进行更多评分。总之,这项研究说明了,在同等交互代价下,更高精度用户自身控制的方法,可以得到更高准确率的偏好模型。

递增的偏好提取方法[1, 5],在一般情况下,也可以用来修正用户的偏好,或价值函数。其他的递增式方法。其他递增方法在特定领域也能改善决策质量。在文献[6, 36, 42]中,研究者通过强调在用户—系统交互的早期展现一组多样性选项的重要性,解决了偏好不确定性的问题。Faltings 等[13, 69, 70]描述了多种刺激用户描述更多的偏好,进而增加决策准确率。Pu 和 Faltings 展示了如何在偏好修正中获得更多的偏好信息,特别是如何解决用户权衡任务[45]。更多的这方面细节请看 16.5.2 节和 16.6 节。

基于经典的和近期的经验性发现,在偏好提取过程中,我们推导出了有关初始偏好提取过程的多项设计指导准则:

**指导准则 1 任意代价:**要考虑新用户偏好的流畅性。允许用户逐步揭示自身偏好。最好是抽取出来和他们最相关的初始偏好,并且选择与他们对于可选项的知识和经验相适应的代价等级。

一个死板的提取过程,通过使用一个事先已经定序的提取过程来获得用户的偏好。当用户被强制地要求在特定顺序或属性上表现偏好时,这与他们真实的意愿是不一致的,他们会错误地得出手段目标,而无法达成根本目标[23]。例如,当用户想要购买机票时,系统可能首先让其选择航空公司,而不是航线。这样,根本目标是要在特定时间飞行的用户不得不得出一个不同的目标,也就是选择航空公司,作为达成根本目标的手段。为了将真实目标转换为手段目标,用户需要得到更丰富的信息,如不同航线的航班时间表。有可能是开始时用户了解的不足,导致其不能得到最优的结果。因此提出准则 2。

**指导准则 2 任意顺序:**用户可以选择任意顺序描述自身的偏好。

如果不能允许用户在足够丰富的属性集合中,描述出其偏好,那么这个提取过程也是死板的。有这么个例子[70],在出行计划系统中,假定用户想在 15 点前达到目的地,可系统只允许用户在检索时设置起飞时间,此时可能用户会错误地认为这次旅程需要转机并且要飞行 5 个小时,那用户就会设置起飞时间为 10 点。其实有个 12 点半起飞的直飞航班 14 点半就能达到目的地,用户就未能搜索到。

手段目标是实现最终决策的中间过程。假定一个用户对于属性  $a_i$  (如到达时间)有个偏好,但是系统只允许用户对属性  $a_j$  (如起飞时间)设置偏好,则用户可能会估算出一个映射关系  $t_i(a_j)$ ,将属性  $a_i$  的值映射到属性  $a_j$  的值(如达到时间=起飞时间+5 小时),然后用户的实际目标  $p(a_j)$  转换为手段目标  $q(a_j)=p(t_i(a_j))$ ,可是如果映射方法是错误的,那手段目标就会导致不准确的结果。



除非属性  $a_i$  与属性  $a_j$  有非常强的关系, 否则实际上是不存在准确的转换关系的。近期对一个购物比较网站([www.pricegrabber.com](http://www.pricegrabber.com))的一次访问表明: 它就没有考虑笔记本电脑的质量属性。可是对于侧重便携性的用户, 质量却是最重要的因素。此时, 当没有质量属性可以参考时, 消费者会将质量关联到大小或是硬盘的容量上。因此, 她往往会检索硬盘空间小和屏幕小的笔记本电脑来替代。这就导致了用户做出了一些不必要的牺牲, 因为许多生产厂家现在可以在适合的硬盘和屏幕尺寸的基础上, 提供质量轻的笔记本电脑。因此我们提出了准则 3。

**指导准则 3 任意偏好:** 允许用户选择表述任意属性上的偏好。

当设计用户交互界面时, 非常有必要的是, 在给予用户最大的控制权限与陷入交互复杂度之间保持很好的平衡。我们推荐使用自适应的交互界面, 让用户能够设定侧重的属性值上偏好, 可以忽略某些并不关注的属性值。另一方面, 在初始查询指定中, 设计者也可以采用逐步上升的方法, 让用户在最初一小部分属性上指明偏好后, 进一步在显示的结果上设置自身偏好。

## 16.5 通过实例激励用户表示偏好

一个有效的提取工具必须能够使用递增的和灵活的方式收集用户的偏好。实践中, 我们也对能够激励用户表述偏好的方法感兴趣。

并非所愿的手段目标常常是因为用户对于当前可用选项不熟悉。同时, 行为理论发现用户在有实例的情况下, 能容易地表述出自己的真实要求[41]。根据 Tversky 的研究[65], 人们虽然不能预先算出偏好顺序, 但是可以从当前选项中做出自己的选择。这些经典的理论也就证明了为什么实例评价在构建偏好时是个很有效的方法[43, 47]。

这一方法被称作实例评价, 是因为用户是通过评价展现给他们的实例产品来构建自身的偏好的。用户初始时可以在任意数目的相关属性上表述自身的偏好。然后, 系统激励用户进行多项被称为“示例和批判”的周期系统让用户进入连续不断的“实例和评价”循环: 展示了一系列实例产品, 让用户进行类似于“我喜欢这款笔记本, 但是想要个硬盘空间更大的”这样的评价。这些评价结果决定了下一个循环需要展示的产品, 当用户找到了想要的产品时, 整个交互过程就结束了。

通过实例评价方法, 用户可以快速地构建出自身的偏好。由于用户只需要做出实例评价, 而不是提供偏好, 整个过程对用户来说, 所需要的代价也更小。更重要的是, 实例评价方法既能满足给用户展示当前已有产品的目标, 也能刺激用户在当前状况下构建偏好。

我们将回顾一整套评价系统, 然后抽象出能够描述这个方法效果的指导原则, 更详细的信息请参考 Lorraine McGinty 和 James Reilly 在本书中第 13 章。

实例评价方法最早出现时, 是作为一项新的数据库访问范型, 特别是给新手用户指明查询要求[72]。近期, 实例评价被广泛使用, 主要有两种形式: 支持产品目录导航和支持基于显性偏好模型的产品检索。

第一类实例评价系统, 如 FindMe 系统[8, 9], 检索被认为是多项搜索的组合, 而浏览则仅用于辅助。这些首先根据用户的初始查询, 展示数据库中最匹配的产品, 然后根据用户的评价重新检索出当前最合适的。按照评价模型实现的界面, 也称为调整, 即允许用户对当前示例做出偏好表述, 如“找一个与这个类似的, 但是周边环境更好的”。根据这一概念, 用户可以调整当前推荐来得到更满意的结果。偏好模型通过当前最佳产品隐式地表示出来, 也就是说, 用户所选的反映了她对于属性值的偏好。在对调整模型改进的基础

上, Reilly 等提出了动态评价模型[53]。除了对单个值的调整操作, 组合评价还允许用户选择与当前最佳物品有两个或更多属性值不同的产品。例如, 初始查询后, 系统给用户推荐了一款数码相机, 以及不同厂家生产的像素更低但空间更大的相机。根据 Apriori 算法[2]产生的组合评价的方法, 允许用户通过更大的跨度, 引导出他们的目标选项。实际上, 在真实用户参与的一项调研[32]中, 频繁使用组合评价的用户能够将他们的交互循环从 29 次降低到 6 次。

第二类实例评价系统, 需要维护一个显性的偏好模型。每个用户的反馈以评价的方式被加入进来细化原始的偏好模型。提供出行计划的 SmartClient 系统就使用了这种模型[43, 63]。一旦用户设置好初始偏好, 它就能提供 30 组实例。当然, 在评价实例的过程中, 用户可能会表述出额外的偏好。这些偏好会积累到一个可以通过接口让用户可见的模型中(参见[64]中图 16.6 中“偏好”下方的底栏), 而且可以随时修改。ATA[28]、ExpertClerk[60]、The Adaptive Place Advisor[16]以及那些递进式动态评价系统的都具有相似的功能[30]。维护一个显性的模型有两个优点: 一是可以避免推荐那些用户排除的产品; 二是可以帮助系统推荐那些用户自己忽略掉的产品, 16.5.2 节将详细地描述。

因此, 为了给用户相关的领域知识, 以及刺激用户构建完整健全的偏好, 我们提出准则:

**指导准则 4:** 给用户示例, 帮助其更流畅地描述偏好。

### 16.5.1 需要多少实例

在设计高效的基于实例的界面时有两个关键点: 一个是需要实例的数目, 另一个是需要展示实例的类型。Faltings 等分析多种偏好模型, 提出了在推荐模型不准的情况[14], 至少需要展示给用户多少实例, 并使得目标选择也被包含进来。他们分析了各种偏好模型。如果用数值效用函数表示的偏好与实际偏好的偏差度是  $\epsilon$ , 而且采用加权总和或最小-最大规则将它们组合起来, 则有

$$t = \left( \frac{1+\epsilon}{1-\epsilon} \right)^d \quad (16.4)$$

其中,  $d$  需要陈述偏好的最大数目,  $t$  是要确保包含目标方案时需要展示的物品数目。由于这个数目与系统物品总数是无关的, 这项技术也可以支持大型数据集的系统。对于一个中等数目的偏好(最多为 5), 最准确实例数目是 5~20。当偏好模型更复杂时, 可能会放大不准确度, 此时需要更多实例。

### 16.5.2 需要哪些实例

最显而易见要展示的实例是那些最能匹配用户当前偏好的实例。然而, 这个策略被证明并不能确保是最优的。由于绝大多数用户并不确定自己的偏好, 所以引导用户构建一个尽可能完整和准确的偏好模型, 对于推荐系统更为重要。可见, 系统需要在用户角度上提供更多的选择。因此我们称被选择用来激励用户的实例为建议。我们给出了两类建议策略: 基于多样性和基于模型的技术。

ATA 系统是最先使用建议的[28], 其实例覆盖了多项属性, 如起飞时间和价格, 也提供了扩展选项, 如更早的和更便宜的。可是, 这些扩展选项对于许多用户来说, 都是没有意义的。例如, 某个用户需要更高分辨率的数码相机, 但并不意味着他可以接受四倍于一般相机的分辨率但也是四倍价格的相机, 实际上, 如果系统进行了这样的推荐, 有可能

会打击到用户, 因为这样暗示了只能通过提高开支来提高性能。

因此, 更好的选择是在那些根据当前已知偏好判断为好的实例中建议, 而且还要强调多样性而不是极端的实例。Bradley 和 Smyth 首先意识到了多样实例的重要性, 特别是对于采用推荐工具的早期阶段[6]。他们提出了边界贪心算法(Bounded Greedy Algorithm), 使得用户能够在检索到与用户查询词最相似结果的同时也能包括最多多样性的结果。因此, 该算法并没有根据偏好  $r(x)$ , 提供  $k$  个最匹配的实例, 而是根据加权总和, 用  $d(x, Y)$  计算实例  $x$  和已选集合  $Y$  的相对多样性:

$$s(x, Y) = \alpha r(x) + (1 - \alpha) d(x, Y) \quad (16.5)$$

其中, 权重  $\alpha$  可以根据最优性和多样性各自重要程度的变化而变化。例如, 当用户得到最终选项时,  $\alpha$  可能会是个很大的值(实验中是 0.75), 这样系统就会强调展示集合的相似度而不是多样性。在他们的实现中, 排序函数  $r(x)$  是  $x$  与理想实例  $t$  在 0 到 1 之间的相似度  $\text{sim}(x, t)$ , 相对多样性可以推导为

$$d(x, Y) = 1 - \frac{1}{|Y|} \sum_{y \in Y} \text{sim}(x, y) \quad (16.6)$$

多样性产生的效果可以根据相对收益来模拟得出, 也就是说, 由于放弃相似度而在多样性方面得到的最大收益[61]。随后, McSherry 等也说明了可以在不牺牲相似度的条件下提高多样性[36]。在排序函数中固定一个阈值  $t$ , 然后在所有的产品  $x$  中找出  $r(x) > t$  就能选出最具多样性的子集。当显示了  $k$  个选项时, 阈值可以是第  $k$  个选项, 这样就能保证相似度不会减少, 或者是某个具体的值, 这样相似度就会有一定程度的减少。

因此我们提出了以下准则:

**指导准则 5:** 给用户提供多样性的实例, 特别是在用户不确定其最终偏好的情况下, 可以刺激用户表述自身偏好。

文献[33]采用的自适应搜索算法, 通过改变排序策略中的  $\alpha$ , 交替采用侧重相似度和侧重多样性这两种策略, 实现了“展示给我更多类似”这样的交互。这种情况下, 当实例产品集展示后, 用户可以选择出他们最喜欢的选项。当用户连续两次选择同一个选项时, 系统在下一轮的实例中, 将会侧重于多样性来提高用户的关注度。否则系统会假设用户还在摸索, 则继续给用户新的选项。有模拟用户参与的评估表明, 这一技术与纯基于相似度的推荐系统相比, 最多可以将推荐循环的长度减少 76%。

最近关于多样性的工作, 主要是针对那些用户偏好不确定的情况[42], 以及覆盖协同过滤推荐系统中不同类型的兴趣[74]。对于一般的偏好模型, 如何定义多样性指标还不是很清楚。Viappiani 等在文献[69, 70, 50]中考虑了当展示建议时, 用户会有意愿说出更多的偏好。一个建议在当前的偏好模型中可能不是最优的, 但是在加入更多偏好后就更有可能是最优的。例如, 某个用户在浏览公寓实例后, 可能会添加“需要带阳台的公寓”这一偏好。9 个不同国籍的 40 位实验主体, 其中 9 位女性, 参与公寓检索的研究。实验结果表明, 使用参考建议不仅可以准确率翻倍, 也能减轻用户 80% 的时间消耗。如果能够得到更好的结果, 用户也愿意给出更多的偏好设置。因此, 他们提出了向前看准则[69, 70, 50]:

**指导准则 6:** 当前偏好模型的建议可能不是最优的, 但是在加入更多偏好后就更有可能是最优的。

向前看准则可应用于构造基于模型的建议, 方法是对于任一属性  $a_i$  都会显式地计算差值度量函数  $\text{diff}(a_i, x)$ , 可以反映在这一属性上偏好能让选项  $x$  最受欢迎的概率。根据物品在所有可能属性上的期望差值度量排序得到:

$$F_a(x) = \sum_{a_i \in A} P_{a_i} \text{diff}(a_i, x) \quad (16.7)$$

其中,  $P_{a_i}$  是指用户被激励去对  $a_i$  属性进行偏好表述的概率。将用户还没有表述偏好的所有属性对应的概率加起来。因此最应该展现的建议就是那些在考虑了隐偏好之后最有可能成为最优的物品。将这些技术综合起来, 可以将最优选项的概率最大化采用这些技术, 就有可能得到一组建议, 使得得到最优物品的概率最大化。详细请看[13, 50]。为了调查这些建议在制定正确决策过程中的重要性, 有人做了一些实际的用户研究[50, 69, 67]。有一项研究里, 用户的行为被在线系统监视着, 研究者记录了 63 位参与者的多轮偏好修正循环。另一项研究, 研究者将 40 位参与者分为两组。一组是基于模型的建议, 一组则不提供建议。这两项研究都说明了使用基于模型建议的效果, 用户在有建议的情况下可以描述更多的偏好([69, 67]监督研究中, 有建议是增加了 2.09 个偏好, 无建议仅增加了 0.62,  $p < 0.01$ ; [69, 50]在线研究中, 有建议是增加了 1.46 个偏好, 无建议是增加了 0.64,  $p < 0.002$ ), 并且用户可以得到更高决策准确率(在监督用户研究[50]中, 有建议是 80%, 无建议仅有 45%,  $p < 0.01$ )。

## 16.6 偏好修正

偏好修正是改变用户之前所说的期望的产品特性, 以及这些特性有多大程度需要满足, 或者是这两者的组合。28 位受试者(10 位女性)被聘来参与从可选的公寓中选出其最喜欢的[48]。用户的偏好可以分为 6 种属性: 类型、价格、面积、浴室状况、厨房状况以及到工作地点的距离。每个参与者首先都得做出选择, 然后使用推荐工具来平衡他们的偏好, 直到选出理想的选项。这个研究中, 在检索过程中, 每个用户都至少修改一个偏好, 许多用户更改偏好是因为几乎没有哪个选项能够满足所有的初始偏好。这里, 最常见的两种偏好修正的情况是: 1) 用户不能找到满足其所有偏好的选项, 不得不选择满足部分的; 2) 用户可能会有很多满足的结果, 不得不减少选项集合的大小。这两种过程都可以看作查询精确化过程, 真正的挑战是帮助用户修改查询参数从而找到目标物品。因此, 我们提出了一个统一的框架, 可以将这两种情况视为权衡过程, 因为有可能不得不在某方面做出牺牲, 来得到可接受的结果。

### 16.6.1 偏好冲突和部分满足

某个用户如果在查询大型公寓时, 要求一个很低的价格, 将会得到“未能找到符合要求”的结果, 接下来他将不知道如何描述更合适的偏好。

当前业界实践都是使用基于浏览的交互技术来处理偏好冲突的情况。用户只能一次提交一次自身偏好, 当他们需要设置更多偏好时, 可以在产品子栏目里设置, 直到在显示的选项中找到目标, 或者再没有产品了。例如, 如果某位用户想要一个很轻的笔记本(轻于 2 千克), 当她设置好质量要求时, 就只能在小于 2 千克的笔记本里进行进一步的搜索。如果这些轻型笔记本的价格非常高, 她有可能会错过那些质量是 2.5 千克但便宜很多的替代选择。这种交互风格在比较性购物网站中很受欢迎(如 [www.shopping.com](http://www.shopping.com)、[www.pricegrabber.com](http://www.pricegrabber.com)、[www.yahoo.shopping.com](http://www.yahoo.shopping.com))。但这种交互是很受限制的, 因为设计者事先已经避免了用户选择冲突的偏好。用户很难设置上下文相关的偏好, 特别是把握好各项属性平衡。如果用户逐步对各项属性进行设置, 匹配的产品可能会突然变为空, 弹出提示“未能找到匹配的物品”。这种情况下, 用户也不知道该修正哪一项属性, 有可能得回退好几步, 才能得到其关注属性的组合结果。

SmartClient 使用了一种更合理的方法[43, 64]。首先, 其允许用户表述自身的所有偏好, 然后给出最大满足部分偏好的选项子集合[15]。这些最大满足的产品可以帮助用户知道当前可选的物品有哪些, 也能让其指定更合理的偏好。基于同样的原理, McCarthy 等提出要通过解释为什么会有这些产品来帮助用户知道有关产品的知识, 而不是为解释系统为什么没能找到满意结果而找借口[31]。FinaMe 系统基于产品类型的背景知识, 在更高层次上[8, 9]解释了偏好集为什么会冲突。当用户想要低耗能和高动力汽车时, FinaMe 系统将会说明这两者是冲突的。这种部分满足的解决方式在 Stolze 的文献[63]中也被称为软导航。

为了让用户相信部分满足的结果, 我们需要采用类似于 activedecision. com 中使用的方法。这种方法里, 不仅展示了部分满足的结果, 也详细解释了系统为什么满足这些用户的偏好, 而不是另外那些。基于一系列的假设和相关问题, 针对这些解释机制, 我们采用精细问卷调查的形式, 制订了一项定性的用户调查。53 位参与者完成了本次调查, 他们中的绝大多数都非常同意解释机制可以更好地激发他们对于推荐系统的信任度[10]。另外, 有一项替代的解释技术, 与传统地将每个物品用一种解释展现相比(见图 16. 2), 将部分满足的产品分组到不同类目里的层次界面更受大多数受试者的欢迎[10]。此外, 还有一项 72 个人参与的用户研究也更深层次地证明了, 界面的方式能在用户的主观感知能力、再次访问的意愿以及节省代价的意愿方面显著激励那些受能力激发的用户的信任度(详情请看 7. 3 节的实验)[49]。

Search Results							
There is NO apartment completely satisfying all your preferences, but							
these apartments are cheaper and bigger, although they are slightly farther							
ID	Type	Price (€)	Area (m²)	Bathroom	Kitchen	Distance (mins)	Basket
27	shared apartment	450	25	private	private	20	Basket
30	room in a house	480	27	private	not available	20	Basket
More							
these apartments are closer and bigger, although they are slightly more expensive							
ID	Type	Price (€)	Area (m²)	Bathroom	Kitchen	Distance (mins)	Basket
77	shared apartment	550	25	private	not available	5	Basket
34	room in a house	600	30	shared	private	5	Basket
More							
these apartments provide private bathrooms, although they are slightly smaller							
ID	Type	Price (€)	Area (m²)	Bathroom	Kitchen	Distance (mins)	Basket
69	shared apartment	470	15	private	shared	10	Basket
72	shared apartment	500	12	private	shared	15	Basket
More							

图 16. 2 展示部分满足用户需求产品的界面

指导准则 7: 在解决偏好冲突时, 可以通过详细解释的方式, 给用户展示部分匹配的结果。

16. 6. 2 权衡辅助

随着目录规模的增长, 找到目标物品变得越来越难。除非有工具帮助用户有效地查看和比较许多潜在有趣的产品, 否则用户的决策精度会很低。尽管推荐系统能够通过过滤和比较功能提高决策质量[20], 但是用户仍然要面对如何选择正确物品的抉择。

我们团队的研究人员发现, 通过对拥有权衡属性的选项进行对比的方式, 可以将决策准确率提高到 57%[48]。28 位受试者(10 位女性)参与了本次实验, 每位参与者首先得做出一个选择, 然后使用决策辅助工具来处理权衡问题。结果证明, 当用户认为一项物品是最终目标时, 推荐工具可以通过给用户一组可以权衡的候选项来得到更高的决策准确度。16. 5 节提出的实例评价界面, 如类似“我喜欢这款便携式电脑, 但有更轻的吗?”同样也能处理这类权



衡问题,这类交互风格称作权衡导航,可以通过类似“修改”组件或“调整面板”(如[47]的图4)来解决问题。FindMe[8]采用的调整技术是最早实现这种权衡辅助的工具。它最早是被设计用来通过每次修改用户已经表述的偏好,帮助用户浏览目标。实例评价方法(SmartClient[48, 47])其实更适合处理这类权衡问题,特别是两项属性的冲突。在单次交互中用户可以陈述他们的想法,提高某个属性值,在其他属性上做出让步,或者两者皆有。

Reilly 等引入了使用动态评价的方法来处理这类权衡问题[53]。为了达到改善系统推荐准确率的目的,评价是在用户可以选择的属性级别上的直接反馈。例如,系统在推荐一款佳能数码相机后,可以给出提示:“更丰富的匹配: 1)更低的光学变焦但是机身更薄更轻的产品; 2)其他厂商像素更低但是更便宜的产品; 3)更大的镜头尺寸和更大内存但是更重。”动态评价是自动产生有用的评价组合的过程,从而用户可以同时指出其在多项属性上的偏好。有实验表明[53],与单个评价相比,动态评价可以减少高达40%的交互周期。

尽管初始时被设计用来解决推荐系统的导航问题,文献[53]描述的单个评价和组合评价相当于文献[47]定义的简单权衡和复杂权衡。这两项机制都可以帮助用户在一组权衡候选集中比较和评估被推荐的物品。然而,动态评价方法提供的是系统提议的权衡支持,这是因为系统在生成和建议权衡的类别,而实例评价提供给用户一种机制,能够初始化他们自己的权衡体验(在文献[11]中被称为用户激发的评价)。

我们团队最近对比了用户激励和系统提议两种方法的效果[11],有36位志愿者(5位女性)参与了这项实验。实验中由受试者参与设计项目,每位参与者都被要求分别评估对应了两种方法的两个界面。评估时用到了2.2节描述的所有三种评估标准:决策准确率、用户交互代价和用户信任度。实验结果表明,用户激励的权衡方法更能提高决策准确率和减少相对的代价,这主要是因为它更灵活,允许用户自由地设置单个和组合评价方式。另外,用户激励的评价方法的信任度也更高,因为这提高了用户购买他们自己寻找到的产品的意愿,以后也将会继续使用系统。因此我们提出:

**指导准则 8:**除了提供检索功能,还需要在界面上为用户提供权衡辅助,可以是系统提议或是用户激励的方式。后者更灵活,也能得到更高的决策准确率和用户信任度。

## 16.7 展示策略

基于偏好的推荐系统至少使用了以下三种展示策略:一次推荐一项物品,展示 $k$ ( $k$ 介于3和30之间)项最匹配的结果,或是通过分值解释的方式展示产品。我们将用ACE框架来讨论这些策略的区别。

### 16.7.1 一次推荐一项物品

这类推荐系统的优点在于,它设计相对简单,用户也不大可能对过剩信息感到惊讶,并且界面也适合类似于手机这类较小的展示设备,但最明显的缺点是用户可能不能快速找到他们的目标。16.5节中提到,新用户的初始偏好是不确定的,因此初始的推荐结果可能没有包含他们的目标选择。并且,由于结果集小,用户不得不与系统进行很多次的交互;或是在还没有找到目标选择前,用户就已经不再继续,这样就导致了更低的决策准确率。因此我们提出:

**指导准则 9:**一次展示一项推荐物品,这种简单策略可能很适合用于小型展示设备,但是,这可能会导致更长的交互周期,或是得到相对低的决策准确率。

16.7.2 推荐  $k$  项最匹配的物品

许多产品检索工具给用户展示了前  $k$  项最匹配的结果，这被称为  $k$  项最匹配界面，如 ActiveDecision.com 用到的商业工具( $k > 10$ )，以及 SmartClient( $7 \leq k \leq 30$ )、ATA ( $k = 3$ ) [28]、ExpertClerk ( $k = 3$ )、FirstCase ( $k = 3$ ) [37] 和 TopCase ( $k = 3$ ) [38] 提到的学术原型。

当  $k$  接近 10 的时候，选项的排序问题变得很重要，最常见的方法是使用同一个评分方法(多属性效用理论[24])，根据用户偏好匹配度对  $k$  项匹配结果排序。我们同样可以使用  $k$  邻域检索算法(或仅仅 kNN) [12] 来排序，如基于案例推理领域用到的方法[25]。这  $k$  个物品，是从高到低，按分值降序的方式排列的(actedecision.com, SmartClient)。该方法可以在不干扰用户的情况下，展示相对多的选项，并且根据用户偏好匹配程度选择，可以得到相对高的决策准确率[48, 47]。

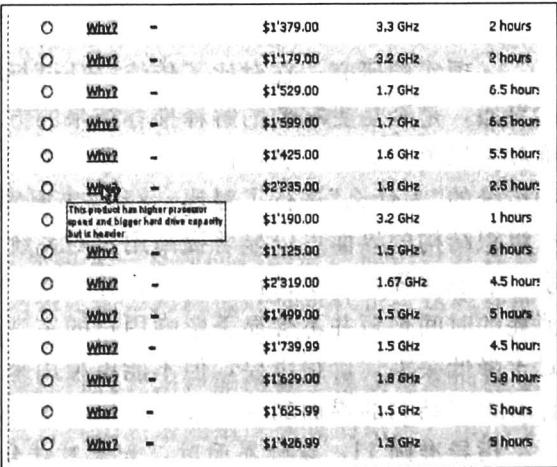
Pu 和 Kumar 做了一项实验，用来对比基于实例评价的系统( $k = 7$ ，根据效用分值排序)和使用排序列表展示方法的系统( $k = n$ ，根据用户选择的属性值排序，如价格) [47, 51]。22 位志愿者参与本次实验，每位志愿者都需要按照给定任务随机顺序的方式，来试用这两种界面。结果表明，在完成这些指定检索任务时，实例评价方法操作更简单(更少的完成时间和更低的错误率)，也能得到相对高的决策准确率[48, 51]，更多的人相信界面的排序结果是正确的。许多用户认为实例评价系统(只展示 7 个物品)隐藏了部分结果，并且界面展示的与实际的排序不一致。从这项实验研究中可以观察到，用户通常不会向下滚动查看更多展示的产品，但他们的信任度更高，而且交互时间没有受到影响。因此我们为  $k$  项最匹配展示策略提出准则：

指导准则 10：通过展示更多的产品，并对其进行自然排序，有可能提高用户的操控感和信任度。

16.7.3 解释界面

提到建议性决策时，如可以选择购买哪款相机，建立用户对系统的信任度并相信推荐结果，这是衡量系统是否成功的关键因素。我们团队的研究者，在普通用户和系统之前的信任关系中，研究了用户信任度和其他主观因素。众所周知，人们对于科技产物(如推荐系统)的信任，也可以抽象出像能力、善意和诚实这些概念，这与人人之间的信任状况是类似的。并且，信任是用户与系统间的一项长久的关系 [10]。如果用户信任一个推荐系统，她更可能会购买推荐物品，并且将会再次光临。在一项有 53 名用户参与的精心设计的定性调研揭示，信任关系的构建才是界面解释其结果的重要能力[10]，见 6.1 节。

解释界面有很多种实现方式。如 ActiveDecision.com 在结果展示界面上添加了一个“Why”标签，在鼠标悬浮在上面时将会看到提示，解释了为什么该项物品匹配了用户的偏好，类似图 16.3 展示的界面。另一方面，也可以设计基于组织结构的解释界面，将最匹配选项展示在最上面，并提供了



<input type="radio"/>	Why?	-	\$1'379.00	3.3 GHz	2 hours
<input type="radio"/>	Why?	-	\$1'179.00	3.2 GHz	2 hours
<input type="radio"/>	Why?	-	\$1'529.00	1.7 GHz	6.5 hours
<input type="radio"/>	Why?	-	\$1'599.00	1.7 GHz	6.5 hours
<input type="radio"/>	Why?	-	\$1'425.00	1.6 GHz	5.5 hours
<input type="radio"/>	Why?	-	\$2'235.00	1.8 GHz	2.5 hours
<input type="radio"/>	This product has higher processor speed and bigger hard drive capacity but is heavier		\$1'190.00	3.2 GHz	1 hours
<input type="radio"/>	Why?	-	\$1'125.00	1.5 GHz	6 hours
<input type="radio"/>	Why?	-	\$2'319.00	1.67 GHz	4.5 hours
<input type="radio"/>	Why?	-	\$1'499.00	1.5 GHz	5 hours
<input type="radio"/>	Why?	-	\$1'739.99	1.5 GHz	4.5 hours
<input type="radio"/>	Why?	-	\$1'829.00	1.8 GHz	5.8 hours
<input type="radio"/>	Why?	-	\$1'625.99	1.5 GHz	5 hours
<input type="radio"/>	Why?	-	\$1'426.99	1.5 GHz	5 hours

图 16.3 典型的带有“Why”标签的推荐系统界面

权衡候选项的多个类别[49]。每个带标签表示的类别都标上了标题，解释了为什么物品会出现在这些类别，如图 16.4 所示。

Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
① -	\$2'095.00	1.67 GHz	4.5 hour(s)	512 MB	80 GB	38.6 cm	2.54 kg
Following products available							
Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○ -	\$1'499.00	1.5 GHz	5 hour(s)	512 MB	80 GB	33.8 cm	1.91 kg
○ -	\$1'739.99	1.5 GHz	4.5 hour(s)	512 MB	80 GB	38.6 cm	2.49 kg
○ -	\$1'625.99	1.5 GHz	5 hour(s)	512 MB	80 GB	30.7 cm	2.09 kg
○ -	\$1'426.99	1.5 GHz	5 hour(s)	512 MB	60 GB	30.7 cm	2.09 kg
○ -	\$1'929.00	1.2 GHz	4 hour(s)	512 MB	60 GB	26.9 cm	1.41 kg
○ -	\$1'595.00	1 GHz	5.5 hour(s)	512 MB	40 GB	26.9 cm	1.41 kg
Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○ -	\$1'220.49	1.8 GHz	5 hour(s)	1 GB	100 GB	38.1 cm	2.95 kg
○ -	\$2'148.99	2 GHz	4 hour(s)	1 GB	100 GB	39.1 cm	2.9 kg
○ -	\$1'379.00	3.3 GHz	2 hour(s)	512 MB	100 GB	43.2 cm	4.31 kg
○ -	\$2'235.00	1.8 GHz	2.5 hour(s)	1 GB	100 GB	43.2 cm	9.99 kg
○ -	\$2'319.00	1.7 GHz	4.5 hour(s)	512 MB	100 GB	43.2 cm	3.13 kg
○ -	\$2'075.00	1.8 GHz	1.67 hour(s)	512 MB	100 GB	43.2 cm	4.4 kg
Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○ -	\$1'529.00	1.7 GHz	6.5 hour(s)	512 MB	80 GB	33.8 cm	1.77 kg
○ -	\$1'599.00	1.7 GHz	6.5 hour(s)	512 MB	80 GB	33.8 cm	1.91 kg
○ -	\$1'125.00	1.5 GHz	6 hour(s)	512 MB	80 GB	30.7 cm	2 kg
○ -	\$2'099.99	1.2 GHz	9 hour(s)	512 MB	60 GB	26.9 cm	1.41 kg
○ -	\$1'649.00	1.1 GHz	9.5 hour(s)	512 MB	40 GB	26.9 cm	1.36 kg
○ -	\$969.00	1.2 GHz	6 hour(s)	256 MB	39 GB	30.7 cm	2.22 kg
Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○ -	\$1'179.00	3.2 GHz	2 hour(s)	512 MB	80 GB	39.1 cm	3.62 kg
○ -	\$1'425.00	1.6 GHz	5.5 hour(s)	512 MB	80 GB	39.1 cm	2.86 kg
○ -	\$1'190.00	3.2 GHz	1 hour(s)	512 MB	80 GB	39.1 cm	3.72 kg
○ -	\$1'629.00	1.8 GHz	5.8 hour(s)	512 MB	60 GB	38.1 cm	3.81 kg
○ -	\$627.10	1.6 GHz	1.5 hour(s)	256 MB	40 GB	38.1 cm	2.81 kg
○ -	\$920.00	1.13 GHz	3.5 hour(s)	128 MB	30 GB	35.8 cm	2.59 kg

图 16.4 更具有激励作用结构化界面

一项实验研究，在对比了传统“Why”标签方式的效果的情况下，探究了类别组织结构的界面，是否是更有效的解释推荐结果的方式。为了弄清楚有组织结构的界面是否是解释推荐结果更有效的方法，由受试者参与设计的一项大规模实验研究，将有组织结构的界面和传统的“为什么”进行了对比。72 位志愿者(19 位女性)参与了这项用户研究。结果证明，有组织结构的界面可以显著提高用户对系统能力的感知度，从而能更有效地提高用户信任度，也能增强其减少代价的意愿，并会再次使用该界面[49]。另外，研究发现，查找产品消耗的时间长短并不会显著影响用户的主观感觉。也就是说，在界面上花的时间少，虽然对于降低决策代价很重要，但不能单独用来预测用户的主观感受。文献[49]提出了高效设计有组织结构的界面的 5 项准则和生成这种界面内容的算法。在此我们提出：

指导准则 11：设计界面时，如果对评分排序的计算过程给出相应解释，将更能提高用户的信任度。

## 16.8 准则验证模型

我们提出了一系列指导准则，来保证在设计产品检索工具时，可以依赖基于以下三项关键标准的普适框架：1) 决策准确率；2) 用户交互代价；3) 用户决策信任度。接下来，我们需要根据用户行为理论，来验证这些准则是否能够优化推荐系统的性能。首先，我们认为产品检索工具必须能够满足大众用户需要，它必须能够适应多种多样的用户，特别是那些为了得到渴望结果，持续地与系统进行交互的用户。在我们的理论模型中，使用  $e$  指代用户的交互代价，由交互步骤来度量。在同样的代价下，如果用户能够每次都得到最匹配的选项，那么用户将会对系统拥有越来越高的信任度。我们将用户认为系统能够达到的信任度，称为感知准确率  $\alpha$ 。

由于每个用户可能会有不同的代价和感知准确率需求，我们可以对用户抽象出三项参数：

- 信任度阈值  $\theta$ ：用户满意搜索过程的当前结果并买下产品所需的感知准确率的数量。
- 代价阈值  $\epsilon$ ：用户为了获得推荐结果所需要的代价。
- 代价增量阈值  $\delta$ ：额外的感知准确率，用来激励用户投入额外的交互循环。

一个设计很差的工具，可能会因为缺乏足够信任度，或在调整交互代价后信任度增值太少，从而失去用户。图 16.5 展示了各个推荐系统中各个步骤的感知准确率，信任度阈值用水平的虚线表示，如果感知准确率未能超过这项阈值，用户将不会购买产品。代价阈值  $\epsilon$  和代价增量阈值  $\delta$ ，可以表示用户能够承受的总代价。在图 16.5 中，用标识“代价界限”的斜虚线表示，当准确率/代价的值小于代价界限时，用户将会放弃整个交互过程。

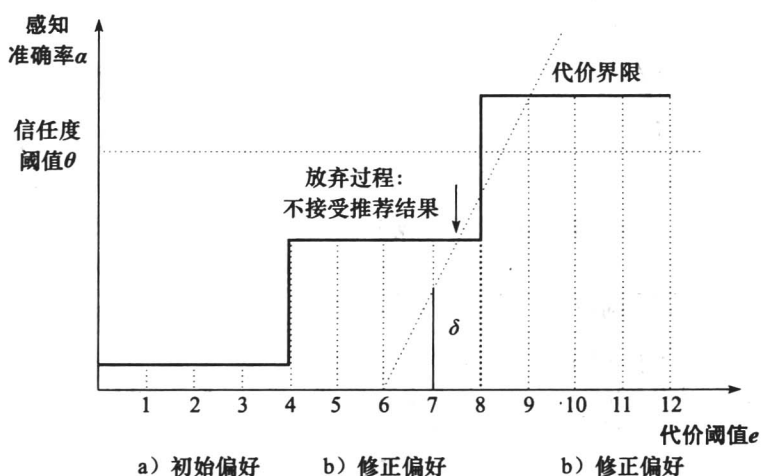


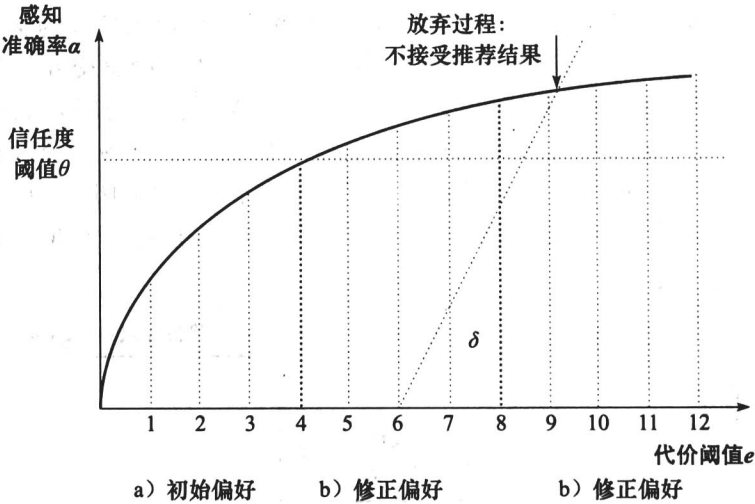
图 16.5 感知准确率与交互代价的关系函数

图 16.5 中使用了一个工具，使得用户可以在包含一系列问题的对话框中设置其自身偏好。很明显，这种界面需要在未能展示任何结果之前，消耗偏好提取代价，这种界面需要在展示结果之前付出很大的代价提取用户的偏好，因此用户的感知准确率会很低，直到用户得到第一批展示结果。紧接着，通过偏好修正过程，用户能够逐步得到更高的感知准确率。这种情况下，当达到信任度阈值时，交互过程仍然未能给出满意的结果，用户将中断整个推荐过程。如果用户有更多的耐心，他们可以在超过信任度阈值和可接受的代价下

得到理想结果，此时曲线到达的代价界限值，是其与“代价界限”的第三次交点。然而，这种情况对用户来说不那么显而易见，因此很难达到这个点。在那些不关注每次代价增加后能显著提升感知准确率的界面设计中，也将会出现这类问题。

为了解决这一缺陷，推荐工具认为感知概率与代价关系函数是一个凹函数，如图 16.6 所示。这个函数利用用户的代价阈值达到了最佳程度：如果感知准确率的增加不足以吸引用户继续使用该工具，用户将不再与系统进行交互。

只有保证对用户的操作即时反馈，并且在交互的早期就采取措施在感知准确率上得到快速增加，才能实现上面所说的凹函数。早期感知准确率的增加，如图 16.6 中的虚线所示，可以让用户在系统中交互更长的时间。基于实例的交互和强调以结构化及激发信任的方法展示多种方案的第 9、10、11 条通用准则，保证了即时反馈。一般情况下，可以假设用户自己会选择增加他们认为能够最大化决策精度的信息，因此用户的意愿是实现该凹函数的关键。阶段 a，系统通过充分挖掘用户的偏好，可以达到最高的准确率。但这个阶段中，如何避免让用户回答那些不能准确回答的问题，是很重要的（指导准则 1）。另外，用户陈述初始偏好时花费的代价越少，这个凹函数曲线将变得越陡，因此我们得使用指导准则 2 和 3。



a) 的准则	b) 的准则	c) 的准则
1. 任何代价; 2. 任何排序; 3. 任何偏好。	4. 展示实例选项; 5. 展示多样实例; 6. 采用预见的原则 推荐选项。	7. 偏好冲突管理; 8. 权衡辅助。
a) ~ c) 的准则 9. 一次展现一个搜索结果适合小屏幕设备，但决策准确率会低; 10. 展示更多的产品并按照自然排序，有可能提高用户的操控感和信任度; 11. 设计界面时，如果对评分排序的计算过程给出相应解释，将更能提高用户的信任度。		

图 16.6 基于示例推荐系统中，感知准确率与交互代价的关系，以及相应的实现指导准则

一旦获得初始偏好后，阶段 b 中，通过让用户完成那些初始未能意识到的偏好，可以得到此阶段最高的感知准确率。可以通过实例激励（指导准则 4）和让用户更了解当前选项



(指导准则 5 和 6)实现,也解决了这一交互阶段的代价和效果的主要冲突。最后,阶段 c 中,通过调整偏好的权重以及做出相应的权衡,能对偏好集进行微调,可以使用部分匹配方案(指导准则 7)和动态偏好权衡的方案(指导准则 8)。

由于推荐工具不能保证用户在阶段转换时(通常是梯度的转换)的效果,需要在每个阶段提供持续的支持,并且鼓励能提高感知准确率的行为。因此,相比添加新的偏好,权重的调整可以不那么显著。

16.5 节中描述的基于实例的推荐系统能更好地解决上述需求,并且通过增加和细化用户偏好,能提高真实的决策准确率。为了保证用户愿意投入整个交互过程中并且愿意接受检索结果,真实准确率也必须能够被用户所感知,可以使用多次展示多项结果(指导准则 9)的方式来纠正误差,可以使用结构化展示界面(指导准则 10)以及结果解释(指导准则 11),这些都是激励用户投入足够的代价来尽可能得到更高实际准确率的基本方法。

## 16.9 总结

本章提出了 11 项关键的指导准则,可以用于设计基于偏好的交互式推荐系统中。为了更好地展示和证明这些准则,我们从深度和广度的角度上回顾之前在侧重用户交互的实例评价方面的研究工作。更重要的是,我们提出了基于三种基本评估标准的框架,这三种标准包括:决策准确率、用户信任度和用户交互代价(简称 ACE)。在该框架里,我们借鉴了已被实践证明的多项技术,来实现上述 11 项指导准则,并且侧重三种标准的结果平衡。通过使用这些准则,可以显著加强产品推荐系统的可用性,也能推广电子商务系统的应用。

## 参考文献

1. Adomavicius, G., Tuzhilin, A., Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17 (6) (2005) 734-749.
2. Agrawal, R., Imielinski, T., Swami, A., Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, ACM Press, 1993, 207216.
3. Belkin, N.J., Croft, W.B., Information filtering and information retrieval: two sides of the same coin? *Communications of the ACM* 35 (12) (1992) 29-38.
4. Bernoulli, D., Exposition of a new theory on the measurement of risk (original 1738). *Econometrica* 22 (1) (1954) 23-36.
5. Blythe, J., Visual exploration and incremental utility elicitation. *Proceedings of the 18th National Conference on Artificial Intelligence*, AAAI press, 2002, 526-532.
6. Bradley, K., Smyth, B., Improving recommendation diversity. *Proceedings of the 12th Irish Conference on Artificial Intelligence and Cognitive Science*, 2001, 85-94.
7. Burke, R., Hybrid recommender systems: survey and experiments. *User Modeling and User-Adapted Interaction* 12 (4) (2002) 331-370.
8. Burke, R., Hammond, K., Cooper, E., Knowledge-based navigation of complex information spaces. *Proceedings of the 13th National Conference on Artificial Intelligence*, AAAI press, 1996, 462-468.
9. Burke, R., Hammond, K., Young, B., The FindMe approach to assisted browsing. *IEEE Expert: Intelligent Systems and Their Applications* 12 (4) (1997) 32-40.
10. Chen, L., Pu, P., Trust building in recommender agents. *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*, 2005, 135-145.
11. Chen, L., Pu, P., Evaluating critiquing-based recommender agents. *Proceedings of Twenty-first National Conference on Artificial Intelligence (AAAI-06)*, 2006, 157-162.
12. Cover, T.M., Hart, P.E., Nearest neighbor pattern classification. *IEEE Transactions on Infor-*

- mation Theory, IT-13, 1967, 21-27.
13. Faltings, B., Pu, P., Torrens, M., Viappiani, P., Designing example-critiquing interaction. *Proceedings of the 9th International Conference on Intelligent User Interfaces (IUI'04)*, ACM Press, 2004, 22-29.
  14. Faltings, B., Torrens, M., Pu, P., Solution generation with qualitative models of preferences. *Computational Intelligence* 20 (2) (2004), 246-263.
  15. Freuder, E.C., Wallace, R.J., Partial constraint satisfaction. *Artificial Intelligence* 58 (1-3) (1992) 21-70.
  16. Goker, M., Thompson, C., The adaptive place advisor: a conversational recommendation system. *Proceedings of the 8th German Workshop on Case Based Reasoning*, 2000.
  17. Goldberg, D., Nichols, D., Oki, B.M., Terry, D., Using collaborative filtering to weave an information tapestry. *Communications of the ACM* (35) 12, Special issue on information filtering (1992) 61-70.
  18. Good, N., Schafer, J.B., Konstan, J.K., Borchers, A., Sarwar, B.M., Herlocker, J.L., Riedl, J., Combining collaborative filtering with personal agents for better recommendations. *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI'99)*, AAAI press, 1999, 439-446.
  19. Ha, V.A., Haddawy, P., Problem-focused incremental elicitation of multi-attribute utility models. In Shenoy, P. (ed.), *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI'97)*, 1997, 215-222.
  20. Haubl, G., Trifts, V., Consumer decision making in online shopping environments: the effects of interactive decision aids. *Marketing Science* 19 (1) (2000) 4-21.
  21. Herstein, I.N., Milnor, J., An axiomatic approach to measurable utility. *Econometrica* 21 (2) (1953) 291-297.
  22. Hill, W., Stead, L., Rosenstein, M., Furnas, G., Recommending and evaluating choices in a virtual community of use. *Proceedings of the CHI '95 Conference on Human Factors in Computing Systems*, 1995, 194-201.
  23. Keeney, R.L., *Value-Focused Thinking: A Path to Creative Decision Making*, Harvard University Press (1992).
  24. Keeney, R.L., Raiffa, H., *Decisions with Multiple Objectives: Preferences and Value Trade-offs*, New York: Wiley (1976).
  25. L, J., Kolodner. *Case-Based Reasoning*. San Mateo, CA: Morgan Kaufmann (1993).
  26. Krulwich, B., Lifestyle finder: intelligent user profiling using large-scale demographic data. *Artificial Intelligence Magazine* 18 (2) (1997) 37-45.
  27. Lang, K., Newseeder: learning to filter news. *Proceedings of the 12th International Conference on Machine Learning*, 1995, 331-339.
  28. Linden, G., Hanks, S., Lesh, N., Interactive assessment of user preference models: the automated travel assistant. *Proceedings of the 6th International Conference on User Modeling (UM'97)*, New York: Springer Wien New York, 1997, 67-78.
  29. Marschak, J., Rational Behavior, Uncertain Prospects, and Measurable Utility. *Econometrica* 18 (2) (1950) 111-141.
  30. McCarthy K., McGinty L., Smyth, B., Reilly, J., A live-user evaluation of incremental dynamic critiquing. *Proceedings of the 6th International Conference on Case-Based Reasoning (ICCBR'05)*, 2005, 339-352.
  31. McCarthy K., Reilly, J., L. McGinty, Smyth, B., Thinking positively explanatory feedback for conversational recommender systems. *Proceedings of the Workshop on Explanation in CBR at the 7th European Conference on Case-Based Reasoning (ECCB'04)*, 2004, 115-124.
  32. McCarthy K., Reilly, J., L. McGinty, Smyth, B., Experiments in dynamic critiquing. *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI'05)*, New York: ACM Press, 2005, 175-182.
  33. McGinty L., Smyth, B., On the role of diversity in conversational recommender systems. *Proceedings of the 5th International Conference on Case-Based Reasoning (ICCBR'03)*, 2003, 276-290.
  34. McNee S.M., Lam, S.K., Konstan, J., Riedl, J., Interfaces for eliciting new user preferences in recommender systems. *Proceedings of User Modeling Conference*, Springer, 2003, 178-187.
  35. McNee S.M., Riedl, J., Konstan, J., Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems (CHI '06)*, ACM, New York, NY, 1097-1101.
  36. McSherry, D., Diversity-conscious retrieval. In, Craw, S., Preece, A. (eds.), *Proceedings of the 6th European Conference on Advances in Case-Based Reasoning*, London: Springer-Verlag, 2002, 219-233.

37. McSherry, D., Similarity and compromise. *Proceedings of the 5th International Conference on Case-Based Reasoning (ICCBR'03)*, Springer-Verlag, 2003, 291-305.
38. McSherry, D., Explanation in recommender systems. *Workshop Proceedings of the 7th European Conference on Case-Based Reasoning (ECCBR'04)*, 2004, 125-134.
39. Mongin, P., *Expected Utility Theory*. Handbook of Economic Methodology, Edward Elgar, 1998, 342-350.
40. Payne, J.W., Bettman, J.R., Johnson, E.J., *The Adaptive Decision Maker*, Cambridge University Press (1993).
41. Payne, J.W., Bettman, J.R., Schkade, D.A., Measuring constructed preferences: towards a building code. *Journal of Risk and Uncertainty* 19 (1999) 243-270.
42. Price, B., Messinger, P.R., Optimal recommendation sets: covering uncertainty over user preferences. *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI'05)*, 2005, 541-548.
43. Pu, P., Faltings, B., Enriching buyers' experiences: the SmartClient approach. *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI'00)*, New York: ACM Press, 2000, 289-296.
44. Pu, P., Faltings, B., Torrens, M., User-involved preference elicitation. *Working Notes of the Workshop on Configuration. Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'03)*, 2003, 56-63.
45. Pu, P., Faltings, B., Decision tradeoff using example-critiquing and constraint programming. *Constraints: an International Journal* 9 (4) (2004) 289-310.
46. Pu, P., Faltings, B., Torrens, M., Effective interaction principles for online product search environments. *Proceedings of the IEEE/WIC/ACM International Joint Conference on Intelligent Agent Technology and Web Intelligence*, 2004, 724-727.
47. Pu, P., Kumar, P., Evaluating example-based search tools. *Proceedings of the 5th ACM Conference on Electronic Commerce (EC'04)*, ACM Press, 2004, 208-217.
48. Pu, P., Chen, L., Integrating tradeoff support in product search tools for e-commerce sites. *Proceeding of the 6th ACM Conference on Electronic Commerce (EC'05)*, ACM Press, 2005, 269-278.
49. Pu, P., Chen, L., Trust building with explanation interfaces. *Proceedings of the 11th International Conference on Intelligent User Interface (IUI'06)*, 2006, 93-100.
50. Pu, P., Viappiani, P., Faltings, B., Stimulating decision accuracy using suggestions. *SIGCHI conference on Human factors in computing systems (CHI'06)*, 2006, 121-130.
51. Pu P., Chen L. and Kumar P., Evaluating Product Search and Recommender Systems for E-Commerce Environments. *Electronic Commerce Research Journal* 8(1-2), June 2008, 1-27.
52. Pu P., Chen L., User-Involved Preference Elicitation for Product Search and Recommender Systems. *AI Magazine* 29(4), Winter 2008, 93-103.
53. Reilly, J., K. McCarthy, L. McGinty, Smyth, B., Dynamic critiquing. *Proceedings of the 7th European Conference on Case-Based Reasoning (ECCBR'04)*, Springer, 2004, 763-777.
54. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J., GroupLens: an open architecture for collaborative filtering of Netnews. *CSCW '94: Conference on Computer Supported Cooperative Work*, ACM, 1994, 175-186.
55. Rich, E., User modeling via stereotypes. *Cognitive Science* 3 (1979) 329-354.
56. Rokach, L., Maimon, O., Averbuch, M., Information Retrieval System for Medical Narrative Reports, Lecture Notes in Artificial intelligence 3055, page 217-228 Springer-Verlag, 2004.
57. Schafer, J.B., Konstan, J.A., Riedl, J., Recommender systems in e-commerce. *Proceedings of the ACM Conference on Electronic Commerce*, ACM, 1999, 158-166.
58. Shardanand, U., Maes, P., Social information filtering: algorithms for automating "Word of Mouth". *Proceedings of the Conference on Human Factors in Computing Systems (CHI '95)*, 1995, 210-217.
59. Schoemaker, P., The Expected Utility Model: Its Variants, Purposes, Evidence and Limitations. *Journal of Economic Literature* 20 (2) (1982), 529-563.
60. Shimazu, H., ExpertClerk: navigating shoppers' buying process with the combination of asking and proposing. *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI'01)*, 2001, 1443-1448.
61. Smyth, B., P. McClave. Similarity vs. diversity. *Proceedings of the 4th International Conference on Case-Based Reasoning (ICCBR'01)*, Springer-Verlag, 2001, 347-361.
62. Spiekermann, S., Parachiv, C., Motivating human-agent interaction: transferring insights from behavioral marketing to interface design. *Journal of Electronic Commerce Research* 2 (3) (2002) 255-285.
63. Stolze, M., Soft navigation in electronic product catalogs. *International Journal on Digital*

- Libraries 3* (1) (2000) 60-66.
64. Torrens, M., Faltings, B., Pu, P., SmartClients: constraint satisfaction as a paradigm for scaleable intelligent information systems. *International Journal of Constraints* 7 (1) (2002) 49-69.
  65. Tversky, A., *Contrasting rational and psychological principles in choice. Wise Choices: Decisions, Games, and Negotiations*, Boston, MA: Harvard Business School Press (1996) 5-21.
  66. Tversky, A., Simonson, I., Context-dependent preferences. *Management Science* 39 (10) (1993) 1179-1189.
  67. Viappiani, P., Faltings, B., V. Schickel-Zuber, Pu, P., Stimulating preference expression using suggestions. Mixed-Initiative Problem-Solving Assistants, *AAAI Fall Symposium Series*, AAAI press, 2005, 128-133.
  68. Viappiani, P., Faltings, B., Pu, P., Evaluating preference-based search tools: a tale of two approaches. *Proceedings of the Twenty-first National Conference on Artificial Intelligence (AAAI-06)*, 2006, 205-210.
  69. Viappiani, P., B. Faltings and Pu, P., Preference-based Search using Example-Critiquing with Suggestions. *Journal of Artificial Intelligence Research (JAIR)*, 27, 2006, 465-503.
  70. Paolo Viappiani, Pearl Pu, and Boi Faltings. Preference-based Search with Adaptive Recommendations. *AI Communications* 21(2-3), 2008, 155-175.
  71. J. von Neumann, Morgenstern, O., *The Theory of Games and Economic Behavior*, Princeton University Press (1944).
  72. Williams, M.D., Tou, F.N., RABBIT: an interface for database access. *Proceedings of the ACM '82 Conference*, ACM Press, 1982, 83-87.
  73. Zhang, J., Pu, P., Performance evaluation of consumer decision support systems. *International Journal of E-Business Research* 2 (2006) Idea Group Publishing.
  74. Ziegler, C.N., S.M. McNee, Konstan, J.A., Lausen, G., Improving recommendation lists through topic diversification. *Proceedings of the 14th International World Wide Web Conference (WWW'05)*, 2005, 22-32.
  75. Zukerman, I., Albrecht, D.W., Predictive statistical models for user modeling. *User Modeling and User-Adapted Interaction* 11 (1-2) (2001) 5-18.

# 基于示意图的产品目录可视化

Martijn Kagie、Michiel van Wezel 和 Patrick J. F. Groenen

## 17.1 简介

传统推荐系统通过排好序的列表向用户进行推荐。例如，在基于内容和知识的推荐系统中，这些列表通常根据与查询、理想产品规范或样本产品的相似度进行排序。然而，该方法会损失大量信息，因为对同一查询具有相同相似度的两种产品可能在另一套完全不同的产品特征集下不再相同。使用基于二维示意图的推荐可视化能够记录部分这样的信息，从而把彼此相似的推荐放在示意图中的同一区域。

基于示意图的方法非常有用的一个领域就是电子商务，因为电子商务商店所卖的大量产品通常由大量的特征所描述，但对某些产品类，这些特征很容易通过产品的图片识别。该方法适用的电子商务领域有电子消费产品和不动产产品。最近业内有人尝试用可视化购物的方法来提升这一类网站的可用性。CrispyShop.com<sup>⊖</sup>就是其中一例，它使用图表来比较产品的特征和价格。另一个可视化购物的方法是根据视觉相似度(如颜色和形状)来选择产品，就像 Like.com<sup>⊖</sup>和 Modista<sup>⊖</sup>做的那样，这在时尚相关领域特别有效。一种与基于示意图的方法更紧密相关的方法已经被 BrowseGoods.com<sup>⊗</sup>采用了，它将产品用类似于百货商店的排序方法进行排序并展示在图中。Musicoverly<sup>⊕</sup>和 LivePlasma<sup>⊕</sup>都展示了一种歌曲示意图，之后他们又创建了类似的电影示意图。最后，Youtube<sup>⊕</sup>有一个称为 Warp! 的想法，推荐和你观看过的电影相似的电影，并把它们展示在图中。

尽管有大量的基于商业示意图的界面，它们却缺少一个公开可用的、科学的基础。在本章，我们将讨论产品目录图界面中的一些问题，这些问题早期发表在[21, 22, 23, 24]。本章结合了这些问题并展示一些在真实电子商务产品目录上的应用。

在 17.2 节中，我们首先回顾一些电子商务领域或相关领域用来创建这些图的科学方法。其中两种，多维缩放(MDS)和非线性主成分分析(NL-PCA)[13, 31, 34]在 17.3 节中会进行详细的探讨，它们通过两种不同的方式来创建产品目录图界面。第一种基于 MDS 的方法使用灵活的相异度指标[21]。第二种基于 NL-PCA 的方法[24]显示了可用于图导航的表示分类属性值(分类属性的可能值)的点。

Martijn Kagie, Michiel van Wezel, Patrick J. F. Groenen, Econometric Institute, Erasmus University Rotterdam, The Netherlands e-mail: martijn@kagie.net, mvanwezel@acm.org, groenen@ese.eur.nl

翻译：何佳倍，燕山大学-王孝先 审核：程文君，胡聪(胡户主)

⊖ <http://www.crispyshop.com>

⊖ <http://www.like.com>

⊖ <http://www.modista.com>

⊗ <http://www.browsegoods.com>

⊕ <http://www.musicoverly.com>

⊕ <http://www.liveplasma.com>

⊕ <http://www.youtube.com>



所有基于图的可视化和基于物品的推荐中的一个共同问题是：不同的产品特征对用户来说并不同等重要。在 17.4 节中，我们介绍一种使用点击流数据来决定属性权重的方法。我们统计产品被卖出的频繁程度。基于这样一种假设：对销量影响大的属性在用户看来就更重要，我们估算了一个泊松回归模型[32, 35]来得到权重。对该方法的描述见文献[22]。

在 17.5 节中，我们描述了一种可将基于图的可视化和推荐系统相结合的方法。该方法结合了 MDS 和文献[45]中的推荐思想。在每一步迭代过程中，我们将包含有限个产品的图展示给用户，用户从中挑选最喜欢的一个产品。接着在下一步中，我们展示和该被选产品更相似的产品。对该推荐系统方法的介绍在文献[23]。

在 17.6 节中，所有上述方法都被用于一个全新真实的 MP3 产品目录。该方法的原型可从 <http://www.browsingmap.com/mapvis.html> 获得。该产品目录由 Dutch 比价网 <http://www.vergelijk.nl> 下的 Compare Group 提供。最后，在 17.7 节中，我们进行总结，并给出了未来研究方向。

17.2 基于图的可视化方法

基于图的方法在用户需要浏览较大数据集的领域中越来越流行，此方法中的相似项目聚在图的相同区域内，如网络搜索、图像浏览和音乐推荐列表管理。同样，基于图的可视化符合业内可视化购物界面的趋势。在这一节，我们讨论上述领域中的 4 种基于图的可视化科学方法，并重点关注用于浏览产品目录的可视化方法及其优缺点。在讨论中，我们特别关注(商业)产品目录中通常包含的数据类型。这些目录通常包含数值属性、类别属性和多值的类别属性。另外，数据中可能存在大量缺失值，例如，不同厂商给出的不同产品说明所造成的。注意，一些可视化方法也能用于三维可视化，但考虑到 3D 可视化对用户过于复杂，我们只讨论 2D 可视化。

为了更深刻理解我们要讨论的 4 种可视化方法之间的差别，我们将它们应用在了一个小的产品目录中。如表 17.1 所示，我们从随后将要用到的 MP3 播放器目录中，选取了流行的 10 个组成该产品目录；此外，为了算法能更容易地执行，我们限定属性数为 4 且属性值都为数值的或二元的；同时，我们选择在 4 种产品属性没有缺失值的产品。

表 17.1 示例数据集

Name	Price	HDD	Memory Size(GB)	Weight
1. Apple iPod Nano	180.55	0	8	40.0
2. Apple iPod Video(30GB)	248.41	1	30	130.0
3. Apple iPod Video(60GB)	73.50	1	60	156.0
4. Apple iPod Video(80GB)	324.00	1	80	156.0
5. Apple iPod Shuffle	76.00	0	1	15.5
6. Creative Zen Nano Plus	76.00	0	1	22.0
7. Creative Zen Vision M(30GB)	199.50	1	30	163.0
8. Creative Zen Vision M(60GB)	250.50	1	60	178.0
9. Sandisk Sansa e280	129.99	0	8	75.2
10. Sony NW-A1200	143.90	1	8	109.0

17.2.1 自组织映射

在网络(搜索)可视化领域中，Kohonen 的自组织映射(SOM)[27, 28, 29] 是创建图

方法中最流行的一种。跟随 Chen 等早期的工作[5]，Chung 等[6，7]和 Yang 等[50]将 SOM 用于实际应用。Ong 等[36]使用 SOM 创建了用于在线新闻的示意图，Van Gulik 等将其用于 MP3 播放器的音乐采集。

自组织图使用非监督神经网络同时进行聚类 and 降维。首先，需要选择一个表示图结构的网格，通常是一个矩形或者六角网格。SOM 训练过程简略地以如下方式运行：所有的网格点(在 SOM 文献中通常称为模型、权重或者原型向量)根据它们在网格中的位置和原始属性空间中的向量来进行初始化。

接着，通过增量学习方法，物品被随机地输入 SOM。对于一个物品，我们首先计算哪一个模型能最好地表示这个物品。该模型称作胜利者或者最佳匹配单元(BMU)，通常使用欧氏距离来确定。然后，对该模型及其邻居进行小程度的调整以适应该物品。对所有的物品进行如上操作后，该迭代结束，下一次迭代开始。在每次迭代中，已有的物品只会受到模型(网格点)及其邻居较小的影响，以便 SOM 收敛到一个解。一种更快的 SOM 批次学习算法见文献[29]。

SOM 的主要优点是它通常能提供聚类效果很好的图，其中的簇由和物品相连的邻居网格点集以及相对较多的空网格点组成。此外，在图中，彼此相似的邻居模型提供了相似程度解释。然而，该相似程度解释对整个图并不总是有效。尽管邻居模型通常更相似，但也有可能在图的另一块中存在着更相似的模型。

另一个缺点是，当两个物品被分配到同一个网格点(模型)中时，它们在图上的同一个位置。虽然可以指定更大的网格或者使用分层 SOM(对每一个原始 SOM 中的模型创一个 SOM)来克服。而且，原始 SOM 不能处理缺失值或者(多值)分类属性，但是通过调整比较指标来确定 BMU 仍是可行的[28]。

图 17.1 为一个基于表 17.1 中样本产品目录的 10×10 的矩形网格 SOM，它使用了 Matlab[1]中的 SOM 工具箱进行创建。如图，每个产品都被分配到方块(网格点)中的一个。整个图都被使用到，即每一个转角都有一个产品被分配到上面，但似乎不是真实的产品聚类。观察图中产品的顺序，垂直维度明显地对应于内存大小和 MP3 播放器是否有硬盘驱动。水平维度捕获另外两种属性。但是因为 SOM 的非线性特征，这些效果并没有在整个图中显示出一致性。

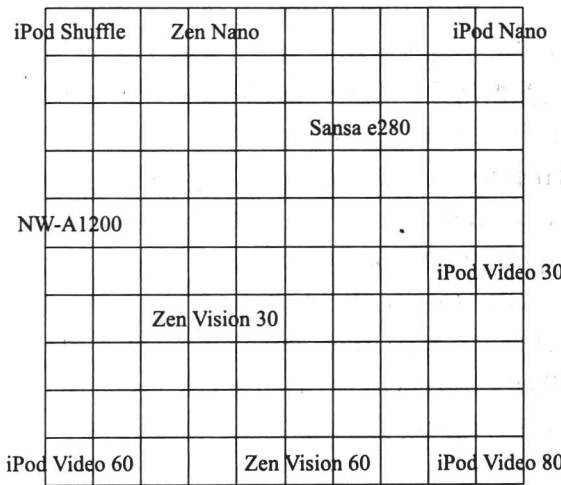


图 17.1 对表 17.1 的样本产品目录的自组织映射示例

17.2.2 树图

另一种用于网络搜索结果可视化[48]和商业应用 NewMap<sup>⊖</sup>的新闻可视化(截图见第 15 章)的常用方法叫作树图算法[44]。

树图使用一棵树，例如，从分层聚类获得的结果作为其输入。每个叶子节点表示一个单独的物品。树中的所有节点都标有权值。对于非叶子节点，其权值为所有子节点权值之

⊖ <http://marumushi.com/apps/newsmapi/index.cfm>

和。当所有叶子节点被赋予权值 1 时，所有节点的权值之和便表示聚类大小。物品也可以通过一些流行度指标进行标记，以便使其在可视化中更为重要。

从根节点开始，我们按照根的子节点个数垂直地将图划分为许多部分，分隔出的结果矩形大小和子节点的权值有关。接着，我们使用相应节点的子节点进一步进行水平划分。交替进行垂直和水平划分，直到触碰到树底并且每个物品在图上都有一个大小与权值相关的矩形。

树图的一个主要优点是整个图都被填充了。图上没有不含任何物品的空间，也没有物品间的重叠。另外，它的分层结构对实现缩放功能非常有用。因为树图可以结合指标相似度的分层聚类算法使用，非常灵活。可以选择一种和已有数据集最匹配的相似度指标。例如，可以使用 17.2.3 节中将介绍的相异度指标来可视化产品目录，它能够处理缺失值和混合属性类型。

树图方法有两个缺点。首先，原始树图算法经常针对小的聚类和单物品产生大的方块，使得单产品类别的可视化很难。该问题可以被正方形化树图[4]或者更好的定量树图部分解决[2]，后者确保代表物品的矩阵有一个确定的长宽比例。第二个缺点是，虽然相似的产品被聚类在一起，但仍没有清楚的距离解释并且簇间的顺序可能丢失，即两个十分相似的产品被分配给不同簇时，它们在图上可能并不近。

一个基于表 17.1 样本产品目录的树图如图 17.2 所示。为了创建该图，我们首先使用 Matlab 基于标准化欧氏距离的平均连接分层聚类算法，并使用 Hicklin 所写的基于 Matlab 的树图算法[18]来可视化结果树。在图上，宽线表示更高的数据划分，等面积的矩阵表示产品。SOM 顶部的产品现在到了树图的右边，这些产品是没有 HDD 的 MP3 播放器。在左边，我们发现相对于 Apple iPod Video 80GB 和 Creative Zen Vision，Apple iPod Video 60GB 更靠近于 Apple iPod Video 30GB、Creative Zen Vision M 30GB 和 Sony NW-A1200（因为从 Apple iPod Video 60GB 划分 Apple iPod Video 80GB 和 Creative Zen Vision M 60GB 的线比下面的线更宽）。

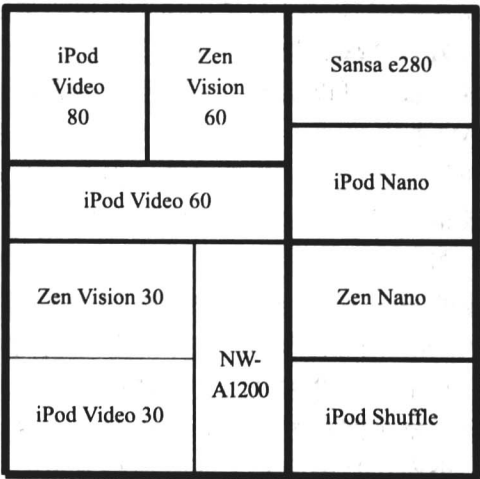


图 17.2 对表 17.1 样本产品目录的树图示例

17.2.3 多维缩放

第三类应用使用多维缩放算法来可视化较大的数据集。已有大量的 MDS 算法[3]如经典缩放[47]（Donaldson 的用于可视化音乐播放列表的算法[11]）、非度量 MDS[30]、Sammon Mapping[42]（Pecenovic 等的可视化图像数据算法[37]）和 SMACOF[10]（Stappers 等的可视化滚轴滑冰目录[46]）。它们都是通过低维欧氏空间绘制一个（对称）的相异度矩阵，使得相异度节点间的距离尽可能靠近。MDS 就是其中一种用于创建产品目录图的算法，我们将在 17.2.4 节进行介绍。

MDS 的主要优点是，图中的距离真实地对应于物品间的相似性。当使用灵活的相异度指标时，MDS 可以处理缺失值和混合属性类。MDS 相对于 SOM 和 TreeMap 的缺点是，大量的空白空间存在于图或图周围，使得图上相似的物品可能会相互覆盖。空白空

间是一种不足, 因为为了能理想地使用图, 对图中特殊部分做更多的缩放是必要的。

图 17.3 为一个使用样本产品目录和 MDS 算法的示意图。我们使用 PROXSCAL 创建该图, 这在使用了 SMACOF 算法的 SPSS 目录下是可行的[33]。相异度矩阵通过标准化的欧氏距离进行计算。如期望的那样, 相对于 SOM, 图的角落更空但图上产品间的位置并没有太大变化。MDS 配置能更好地匹配相异度, 但另一方面, 它使得图更缺乏组织。注意, 在大多数情况下二维里不可能完美地匹配所有相异度, 因此这种让部分相异度更好的解决方法只是一种折中。

#### 17.2.4 非线性主成分分析

主成分分析(PCA)也许是用于降维和创建图最著名的方法。PCA 用数值型数据

矩阵作为输入进行线性变化, 使水平绘制的第一维尽可能多地代表数据中的变化量。后面的维(假设只有二维)也试着表示剩余的变化量。其中, 两两维度不相关。非线性主成分分析(NL-PCA)[13, 31, 34]相对于 PCA, 还能解决分类属性和缺失值的问题。通过序数变换, 数值属性可以非线性转换。此外, 在图上属性分类也具有了一个位置, 即属于分类的物品集的中心。这些分类点既可用于解释图, 也可以在图的界面上作为一种导航的工具, 就像我们的产品目录图采用 NL-PCA 所做的那样, 这将会在 17.2.5 节中介绍。

图 17.4 展示了样本产品目录的 NL-PCA 配置, 它同样使用了 SPSS 目录下的 CATPCA 方法[33]。其中, 所有的数字属性都被当作序数值, 使得非线性转换变为可能。此外, 由于 NLPCA 的灵活性和小数据集, 生成的图有大量的重叠产品。事实上, NL-PCA 创建了非常清晰的三个簇。虽然其他三种方法产生的图中的簇也比较靠近, 但它们毕竟没有形成真正的簇。该例说明, NL-PCA 聚类产品的能力在极端情况下有相同的分类值。

### 17.3 产品目录图

本节介绍两种创建产品目录的示意图的方法。一种基于多维缩放的产品目录图, 它结合了 k-means 聚类算法来突出一些原型产品; 另一种基于非线性主成分分析(NL-PCA), 它利用图中目录点来提供导航机制。这两种方法早期发表的文章见[21]和[24]。

我们在该方法及后面使用以下数学符号: 产品目录  $D$  包含  $I$  个产品, 产品  $i$  包含  $K$  个属性  $x_i = (x_{i1}, x_{i2}, \dots, x_{iK})$ 。最终的二维图可用包含图中产品坐标的  $I \times 2$  矩阵  $Z$  描述。

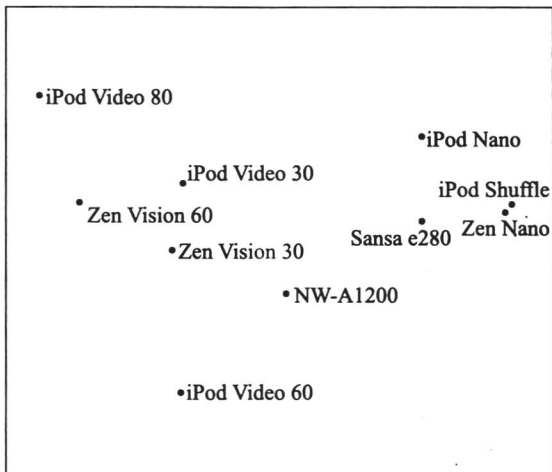


图 17.3 对表 17.1 中展示产品目录进行多维缩放得到的示意图的示例

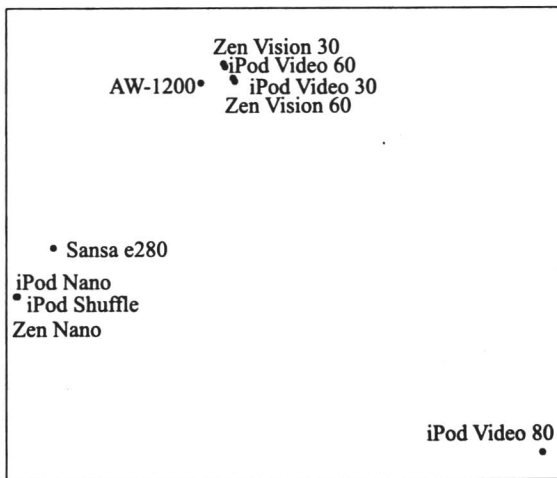


图 17.4 对表 17.1 中展示产品目录进行非线性主成分分析得到的示意图的示例

### 17.3.1 多维缩放

我们描述的第一种创建产品目录图的方法是基于 MDS 的。17.2.3 节中提到的 MDS 图的基础是一个相异度矩阵。为了从产品目录中计算相异度矩阵  $\Delta$ ，我们需要一种相异度评价指标。该指标能够处理产品目录中的特殊数据，即它能够处理缺失值、数值的、分类属性和多值分类属性。

很多流行的相似(异)度指标，如欧氏距离、皮尔逊相关系数以及 Jaccard 相似度指标，不能够处理所有上述属性类。并且，它们也不能自然地处理缺失值。因此，我们使用相异度指标作为文献[23]中 Gower[14]提出的相似度系数的改版。请注意，基于 MDS 的产品目录图方法也能够使用其他相异度指标，如共同购买关系或者物品-物品关系。

产品  $i$  和  $j$  的相异度  $\delta_{ij}$  定义为非缺失相异度评分在  $K$  个属性下的均值的平方根

$$\delta_{ij} = \sqrt{\frac{\sum_{k=1}^K w_k m_{ik} m_{jk} \delta_{ijk}}{\sum_{k=1}^K w_k m_{ik} m_{jk}}} \quad (17.1)$$

其中  $w_k$  是属性  $k$  的权值， $m_{ik}$  和  $m_{jk}$  是二元符号，当产品  $i$  或  $j$  的某个属性值  $k$  缺失时为 0，反之为 1。权重  $w_k$  可使某些属性更重要。17.4 节中，我们将展示自动地分配这些权值来匹配用户的个人偏好。相异度分数  $\delta_{ij}$  基于属性  $k$  的类，对所有属性类，我们使用相同方式的标准化来保证产品目录中每一个属性的平均非缺失相异度分数等于 1。该标准化的作用是使相异度分数同等重要并且独立于缺失值数。

数值相异度分数基于绝对距离

$$\delta_{ijk}^N = \frac{|x_{ik} - x_{jk}|}{\left(\sum_{i < j} m_{ik} m_{jk}\right)^{-1} \sum_{i < j} m_{ik} m_{jk} |x_{ik} - x_{jk}|} \quad (17.2)$$

分类属性相异度分数

$$\delta_{ijk}^C = \frac{1(x_{ik} \neq x_{jk})}{\left(\sum_{i < j} m_{ik} m_{jk}\right)^{-1} \sum_{i < j} m_{ik} m_{jk} 1(x_{ik} \neq x_{jk})} \quad (17.3)$$

其中  $1()$  是符号函数，当条件成立时返回 1，反之返回 0。

为了处理一个产品可以属于多个分类的分类属性(多值分类属性)，相异度框架在文献[22]中被扩展。对多值分类属性的相异度分数定义为

$$\delta_{ijk}^C = \frac{|x_{ik} \Delta x_{jk}|}{\left(\sum_{i < j} m_{ik} m_{jk}\right)^{-1} \sum_{i < j} m_{ik} m_{jk} (|x_{ik} \Delta x_{jk}|)} \quad (17.4)$$

其中  $x_{ij}$  和  $x_{jk}$  是值集， $\Delta$  是对称差集操作符。该指标统计了有多少个目录包含和不包含一个产品。

17.2.3 节提到的 MDS 方法的目标是，使相异度矩阵  $\Delta$  (具有元素  $\delta_{ij}$  并且在我们的方法中使用式(17.1)进行计算)尽可能地匹配一个低维欧氏空间中点与点之间的距离。该目标可以通过最小化原始压力函数[30]进行规范化：

$$\sigma_r(\mathbf{Z}) = \sum_{i < j} w_{ij} (\delta_{ij} - d_{ij}(\mathbf{Z}))^2 \quad (17.5)$$

式中， $\mathbf{Z}$  是一个  $I \times 2$  的坐标矩阵，作为图的基础， $\delta_{ij}$  是物品  $i$  和  $j$  之间的相异度， $d_{ij}(\mathbf{Z})$  是  $i$  和  $j$  坐标的欧氏距离，即



$$d_{ij}(\mathbf{Z}) = \sqrt{\sum_{s=1}^2 (z_{is} - z_{js})^2} \quad (17.6)$$

同时, 通过指定权重  $w_{ij}$  可使某些相异度能更好地匹配。

我们使用的相异度指标能够处理缺失值。然而, 基于只有几个(甚至都不重要)的非丢失属性的相异度要比相异度分数没有丢失的相异度要更不可信, 因此后者应该接受更高的权重。我们可以通过定义式(17.5)中的权重作为  $ij$  对非丢失属性的加权平衡。

$$w_{ij} = \frac{\sum_{k=1}^K w_k m_{ik} m_{jk}}{\sum_{k=1}^K w_k} \quad (17.7)$$

我们使用 SMACOF 算法[10]最小化  $\sigma_r(\mathbf{Z})$ 。该方法的一个优点是它在理论上更快并且迭代产生单调递增的压力值, 并且子结果坐标矩阵  $\mathbf{Z}$  中的差收敛到 0[10]。该性质对动态可视化有一个重要和关键的影响: 算法通过平滑改变的点产生一个(局部)最小解式(17.5), 实际上该物品在屏幕上遵循一个光滑的轨迹。

当产品数量过多时, 生成的图可能对用户看起来过于庞大。为了使图对用户更有吸引力, 一些产品通过大的全彩图来强调, 而其他产品使用小一点的单色图。这些被强调的产品能帮助用户更快地得到一个图的全貌。通过使用 k-means 方法聚类图中的产品可以实现这一点[17]。我们决定在图  $\mathbf{Z}$  上执行 k-means 聚类而不是在原始的相异度上执行分层聚类, 是考虑到两点原因: 第一, 该过程更快; 第二, 它与可视化更一致, 即图中没有重叠。每一个簇中, 与簇中心欧氏距离最近的产品被选择强调。17.6 节展示一种用于 MP3 播放器的产品目录的该方法原型。

### 17.3.2 非线性主成分分析

第二种创建产品目录图的方法是基于 NL-PLA。该方法创建的图中, 不只是产品, 属性分类值也被画出, 从而可以用于导航和选择。NL-PCA 是对序数(非线性排序)分类属性的标准主成分分析的泛化。当只有数字属性时, NL-PCA 就简化为标准 PCA; 当所有属性值都一样并且选择一个多标称变换时, NL-PCA 就等价于齐次或多重一致性分析。

在齐次分析中, 通过所有属性的符号矩阵  $\mathbf{G}_k$  对  $I \times K$  数据矩阵  $\mathbf{X}$  建模。令  $L_k$  表示属性  $k$  的目录数, 每个目录  $\ell$ ,  $\ell=1, \dots, L_k$  在  $I \times L_k$  矩阵中  $\mathbf{G}_k$  有其自己的列, 其中 1 表示该物品属于该目录, 0 表示否。多值异样属性使用  $I \times 2$  符号矩阵对每一个目录建模。缺失值使用一个  $I \times I$  二元对角矩阵  $\mathbf{M}_k$ , 对所有属性, 丢失则值为 1, 反之为 0。使用该数据表示, 我们可以对齐次分析定义如下损失函数[13, 34]

$$\sigma(\mathbf{Z}; \mathbf{Y}_1, \dots, \mathbf{Y}_K) = K^{-1} \sum_{k=1}^K \text{tr}(\mathbf{Z} - \mathbf{G}_k \mathbf{Y}_k)' \mathbf{M}_k (\mathbf{Z} - \mathbf{G}_k \mathbf{Y}_k) \quad (17.8)$$

其中  $\mathbf{Z}$  是一个  $I \times 2$  矩阵, 表示二维欧氏空间中的物品; 矩阵  $\mathbf{Y}_k$  是属性  $k$  的目录值的二维表示。物品坐标  $\mathbf{Z}$  和属性目录值坐标  $\mathbf{Y}_k$  都可以绘制在一个叫作双标图[15]的交叉空间中。本质上,  $\mathbf{Z} - \mathbf{G}_k \mathbf{Y}_k$  表示单个产品的位置及其所属的目录中心的差(或误差)。理想的情况是, 没有差值存在并且所有属于同一目录的产品都属于同一类并和该目录重叠。由于属性很多, 产品在不同的属性下会填充到不同的分类, 式(17.8)简单地通过产品与其所属分类中心的距离平方来衡量均匀性。矩阵  $\mathbf{M}_k$  移除了所有包含丢失属性值  $k$  的产品对误差的影响。式(17.8)可用称作同质分析[13, 34]的可选最小平方(ALS)过程进行最小化。

同质分析对 NL-PCA 里的分类点进行了简单的额外限制。要将数值和序数属性整合到同质分析框架, 可以通过加一个 rank-1 限制

$$Y_k = q_k a_k' \quad (17.9)$$

给数值和序数属性, 其中  $q_k$  是一个  $L_k$  维分类量化列向量,  $a_k$  是一个二维权值列向量。使用该限制分类点被排成一条线。但是, 该限制不足以保存序数属性的顺序甚至数值属性的相对距离。因此, 每个 ALS 迭代中都对  $q_k$  进行约束以满足前述限制。序数属性通过加权单调回归进行变换[9], 至于数字属性, 则通过用属性的序数值  $x_k$  替代  $q_k$ 。关于 NL-PLA 的 ALS 算法的详细描述在[13, 31, 34]。

NL-PCA 的解的一些优点使得它更适于创建既包含产品也包含属性分类值的图。

- NL-PCA 提供一个物品点和属性分类点的交叉空间, 称为双标图[15], 而其他可视化方法则只提供物品点。分类点可用于为图提供更简单的解释和通过选择产品子集导航整个图。
- 对于分类属性, 分类点处于所属物品点的中心。这意味着当某个属性在图中充分体现出来时, 对象被聚类在该属性相应的分类值周围, 选择所有属于该分类的点就选出了图上的一个子空间。序数属性的目录点将是线上最接近物品点中心的点。
- NL-PCA 和大多数其他可视化方法的第三个优点是, 图中物品点的距离通常相关于它们的相异度。

17.6 节也将展示一个该方法在 MP3 播放器上的应用。

## 17.4 通过点击流分析决定属性权重

17.3 节介绍的两种产品目录图的生成方法, 都认为所有属性对用户同等重要。然而, 我们通过 MDS 方法的相异度指标整合属性权值; 同样 NL-PCA 也可以调整以整合不同的属性权值。它适用于大部分可视化方法, 包括 17.2 节介绍的自组织图和树图。

本节将介绍一种根据点击流数据自动决定这些权值的方法。我们统计每个产品在一段时期被卖出的次数。在 17.6 节展示的应用中, 我们实际上统计了跳出率, 即跳出一个可购物商店(我们使用一个比价网的数据)的点击。为了更容易阅读, 我们在后面使用术语销量而不是跳出率。根据这些统计和产品属性, 我们估计了一个泊松回归模型, 该模型属于通常的线性模型。根据该模型的系数和相应的标准误差, 我们计算了  $t$  值, 作为我们属性权值的基础。该方法早期描述于文献[22]。

### 17.4.1 泊松回归模型

一些常常用于统计学领域的模型是广义线性模型(GLM)[32, 35]。为了建模一个离散非负的独立计数变量, 如销量, 我们使用 GLM 家族中的一个合适的成员, 即泊松回归模型。在泊松回归中, 我们不能直接使用(多值)分类属性, 所以不得不创建虚拟属性进行替代。每个虚拟属性用  $L_k - 1$  个虚拟值  $x_{ikl}$  表示, 当物品属于该分类为 1, 反之为 0, 其中  $L_k$  是属性  $k$  的不同分类的数量。当一个物品属于最后一个分类  $L_k$  时, 所有该属性的虚拟值都为 0。选择这种表示方法是为了避免多重共线性。对多值分类属性使用同样的方法, 只是现在所有分类都用  $L_k$  个虚拟值表示。对于数值属性, 我们就使用该属性本身。因此  $x_{ik} = x_{ik1}$ ,  $L_k = 1$ 。我们得到物品  $i$  在向量  $x_i$  中的所有  $x_{ikl}$ 。一个截距项  $x_{i0}$  也被包含在该向量中, 对所有物品, 其值为 1。因此,  $x_i = (x_{i0}, x_{i1}, \dots, x_{iKL_k})$  有了对所有  $I$  个物品的独

立的统计变量值  $y_i$ ，我们就可以如下表示泊松回归模型

$$y_i \approx \exp(\mathbf{x}_i' \mathbf{b}) \quad (17.10)$$

其中  $\mathbf{b}$  是待估回归参数向量，进一步假设  $y_i$  服从期望  $E(\exp(\mathbf{x}_i' \mathbf{b}))$  的泊松分布。该泊松回归模型可以通过最大似然度函数进行拟合，通常使用迭代加权最小二乘法实现。除了模型参数  $b_k$ ，标准差  $\sigma_k$  也可以通过该法得到。

#### 17.4.2 处理缺失值

遗憾的是，泊松回归模型不能同时处理缺失值。缺失值是 GLM 中的一个常见问题，IMbrahim 等[20]最近比较了处理 GLM 中缺失值的不同技术。其中一种最好的方法(产生无偏估计和可信标准差)叫多重填补(MI)[41]。MI 方法根据非缺失值的分布来填补缺失值以得到一组“完整”数据集。这些虚拟数据集可用两种方法创建：数据扩充[43]和重要性采样/重采样[26]。虽然这些方法的结果质量都差不多，但是第二种方法计算速度更快。因此，使用第二种方法的算法——Amelia 算法[26]，作为一个统计软件环境 R 的包，用于我们的方法。我们在文献[22]中讨论了如何用虚拟数据集的回归系数和标准差评估估计泊松回归模型的参数和标准差。

#### 17.4.3 试用泊松回归选择权值

有三个理由使我们不能直接使用从泊松模型得到系数  $b_k$  作为权重。第一，相异度分数并且属性的数量级不同。第二，当直接使用  $b_k$  时，没有考虑系数正确性的不确定性。即使系数值较大，它仍然可能不重要。例如，考虑一个有着很少 1 和很高系数值的虚拟属性，由于它的高系数只用在了有限个物品上，限制了它的总体重要性。我们可以通过将不确定性纳入计算来对此进行校正。第三，权重应该总大于等于 0 的，但  $b_k$  可能为负。

$$t_k = \frac{b_k}{\sigma_k} \quad (17.11)$$

前两种问题可以使用系数  $b_k$  的  $t$  值作为计算权值的基础。通过标准化，所有  $t_k$  都处于同一数量级。并且，由于以标准差  $\sigma_k$  作为分母，不确定性也被考虑了进来。当使用  $|t_k|$  而不是  $t_k$  时，我们确保权值大于等于 0。

对于数值属性，我们可以把  $|t_k|$  直接映射到一个“伪”绝对  $t$  值  $v_k$ ，即对属性  $k$ ， $v_k = |t_k|$ 。有了权值的标准化(为了便于解释)，我们就可以计算  $w_k$

$$w_k = \frac{v_k}{\sum_{k'=1}^K v_{k'}} \quad (17.12)$$

对于(多值)分类属性，首先必须使用  $t_k$  的  $L_k$  值计算  $v_k$ ，它等于  $t_k$  的平均绝对值

$$v_k = \frac{1}{L_k} \sum_{\ell=1}^{L_k} |t_{k\ell}| \quad (17.13)$$

进而结合式(17.12)和  $v_k$  可以计算多值分类属性的权值。

#### 17.4.4 阶梯式泊松回归模型

$t$  值可以同  $t$  分布比较来判断是否和 0 显著不同。在所谓的逐步模式中，使用该显著性测试选择属性，最终生成一个只包含显著属性的模型。该逐步模型方法从一个包含所有

属性的模型出发，每一次从中删除最不显著的属性，直到该模型只包含显著属性。注意，这种逐步方式在同时删除所有不显著属性时会导致不一样的模型。事实上由于共线性，当某个属性被删除时，其他属性的显著性也可能改变。当使用逐步模型决定权重  $w_k$  时，我们将虚拟值个数  $L_k$  考虑到最终的模型中。17.6 节将展示使用该逐步模型的应用。

## 17.5 图像购物界面

基于图的可视化思想可以结合推荐系统[39]。最直接的方法是将推荐系统的结果用一个图来展示而不是列表。这种方法在[23]中的图像推荐系统被用到。该论文还介绍了另一个系统，图像购物界面(GSI)。当用户不清楚自己想要找什么并且需要明确他们的爱好时，GSI 可以提供很好的交互式推荐。GSI 实现了一种在推荐系统中叫作提议推荐[45]或者灵感搜寻[40]的方法。其想法是使用户一步步浏览整个产品目录，每一步中只有部分产品被呈现在图上。用户可从该图中选择某个产品，接着和该产品更相似的产品通过 MDS 生成和显示，该方法和 17.3.1 节基于 MDS 的产品目录图类似。

文献[23]提议了三种不同类的 GSI，即随机系统、聚类系统和分层系统。因为随机系统在该文献的模拟研究中表现最佳，所以在此只考虑随机系统。

该系统中，每一步从较大集中随机选择小部分产品展现给用户，该较大集包含与用户所选产品相似的产品。首先，GSI 需要初始化，我们以该初始化作为迭代  $t=0$ 。在该初始化中，较大产品集  $D_t$  是完全产品目录集合，即  $D_0=D$ 。从  $D_0$  中随机选择(无替代) $p$  个产品作为输出，组成小集合  $D_0^*$ 。然后根据该  $D_0^*$  和合适的 Gower 系数计算相异度矩阵  $\Delta_0^*$ ，Gower 在 17.3.1 节中提到过。最后，使用 MDS 得到的随机产品呈现在图  $Z_0$  上，并将该图呈现给用户。

当用户选择一个展示产品时，迭代过程将开始，标记这个产品为  $x_i^*$ 。接着，计算  $x_i^*$  和  $D$  中其他所有产品的相异度，选择和  $x_i^*$  最相似的  $\max(p-1, \alpha^t I - 1)$  个产品组成  $D_t$ 。参数  $\alpha (0 < \alpha \leq 1)$  决定每步迭代中  $D_t$  的减少量。之后的过程和初始化几乎一样。我们从  $D_t$  中随机选择  $p-1$  个产品和  $x_i^*$  组成小集  $D_t^*$ ，并计算相异度矩阵  $\Delta_t^*$ 。插 U 那个键一个由  $x_i^*$  和  $p-1$  个从  $D_t$  中随机选择的产品组成的小集合  $D_t^*$ ，并计算相似度矩阵作为 MDS 算法的输入，并返回新图  $Z_t$ 。

参数  $\alpha$  决定用户整个选择过程的影响程度。当  $\alpha=1$  时，这个影响非常小，因为除了最后一次迭代用户选择的产品，每次迭代都是完全随机选择产品展现给用户； $\alpha$  较小时，影响较大，但是  $D_t$  方差减少也更快。图 17.5 对这个随机系统进行了概述。

```

procedure RANDOMGSI( $D, p, \alpha$ )
   $D_0 = D$ .
  Generate random  $D_0^* \subset D_0$  with size  $p$ .
  Compute  $\Delta_0^*$  given  $D_0^*$  using (17.1).
  Compute  $Z_0$  given  $\Delta_0^*$  using MDS.
   $t = 0$ .
  repeat
     $t = t + 1$ .
    Select a product  $x_t^* \in D_{t-1}$ .
    Get  $D_t \subset D$  containing  $\max(p-1, \alpha^t I - 1)$  products most similar to  $x_t^*$  using (17.1).
    Generate random  $D_t^* \subset D_t$  with size  $p-1$ .
     $D_t^* = D_t^* \cup x_t^*$ .
    Compute  $\Delta_t^*$  given  $D_t^*$  using (17.1).
    Compute  $Z_t$  given  $\Delta_t^*$  using MDS.
  until  $D_t^* = D_{t-1}^*$ .
end procedure

```

图 17.5 图形购物界面伪代码

在 17.6 节也展示 GSI 的一个应用。

17.6 电子商务应用

本节介绍三种在线应用原型，有两种是基于 MDS 和 NL-PCA 来应用于产品目录图的，第三种原型是 GSI(graphical shopping interface)。MDS 和 GSI 都使用了 17.4 节中的方法来决定权值。所有原型都使用下面的 MP3 播放器产品目录。

MP3 播放器产品目录由 Compare Group 提供，其拥有欧洲比价网站中的 Dutch 比价网 <http://www.vergelijk.nl>。我们所用的产品目录是来自该网站 2007 年 10 月的数据库备份。当时，该网站有 225 款 MP3 播放器，这些播放器用 45 种属性描述，其中 16 种数值的 20 种分类的和 45 种多值分类。在这 45 种属性中，有 26 种属性值的缺失值超过了一半，不同于[21, 23, 24]中使用的。为了能够如 17.4 节中所述的自动决定权值，我们将 2007 年 7 月 15 日到 9 月 15 日这 2 个月的产品目录和相同网站的点击流日志进行了匹配。

表 17.2 MP3 播放器数据的属性特征。这里只列出了缺失值少于 50% 的属性。  
对(多值)分类属性来说，只显示 3 个最常见的值

	属性	缺失值/%	均值
数值属性	Price	0.0%	134.54
	Height	40.9%	63.65
	Width	40.9%	48.87
	Weight	44.0%	71.21
	Depth	40.9%	16.54
	Memory Size	3.1%	8635.30
	Battery Life	44.4%	17.26
分类属性	Brand	0.0% Samsung(12.0%), Creative(9.8%), Philips(8.4%)	
	Radio	32.0% yes(68.6%), no(31.4%)	
	Extendable Memory	44.9% yes(17.7%), no(82.3%)	
	Equalizer	39.6% yes(85.3%), no(14.7%)	
	Screen	30.2% yes(99.4%), no(0.6%)	
	Battery Type	44.4% li-ion(44.8%), 1×AAA(33.6%), li-polymer(20.8%)	
	Voice Memo	24.4% yes(81.2%), no(18.8%)	
多值分类属性	Storage Type	42.7% flash memory(69.0%), hdd(21.7%), sd card(10.1%)	
	Interface	4.0% usb(63.4%), usb 2.0(31.5%), hi-speed usb(7.4%)	
	Color	38.2% black(68.3%), white(20.1%), silver(16.5%)	
	Operating System	31.1% windows(78.7%), mac os(34.2%), windows xp(32.3%)	
	Audio Formats	1.8% MP3(98.6%), WMA(90.0%), WAV(48.0%)	

17.6.1 使用属性权值的基于 MDS 的产品目录图

基于 MDS 的产品目录图原型的 JAVA 应用可从 <http://www.browsingmap.com/mapvis.html> 获得，这个原型展示见图 17.6。该原型的 GUI 主要由一个大的焦点图组成，该焦点图包含了完整图中部分产品的详细视图。该图中，一些产品(本例中为 12)用较大的全彩图展示，其他产品则用较小的单色图。单色图默认根据它们所属的簇着色。用户可以



根据产品的流行度(基于点击流数据)或者属性值着色。顶部小图总是展示整个产品目录图。用户可以以多种方式浏览整个目录图。通过在焦点图或全览图中选择一个矩形子空间,用户可以在图的特定部分进行缩放,也可以通过鼠标滚轮方式完成。用户还可以通过拖拽在图上移动。由于图本身是静态的,即产品坐标是固定的,图的计算可以离线进行,以减少在线计算时间。当然,直接的方法是加入传统的搜索技术,例如,在图的界面中能执行明确的查询,剩下的所有满足约束的产品可用图展示。这要求当产品变化时图随之发生调整,所以客户端的计算可能是必要的。

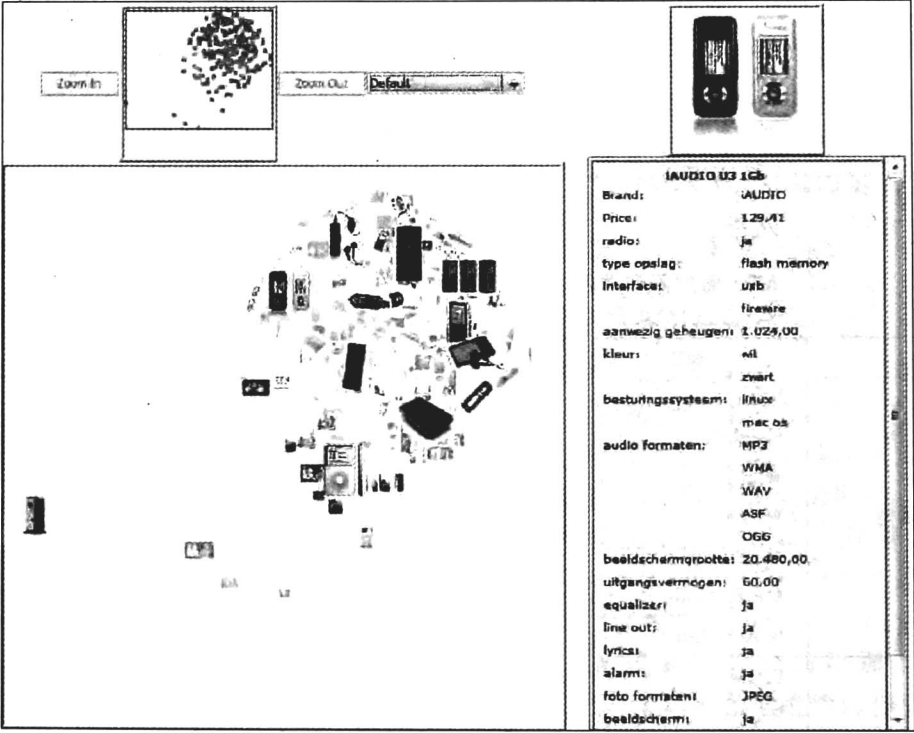


图 17.6 基于 MDS 的产品目录图原型的截图

原型图通过 17.4 节中的属性权值确定技术来实现。这些权值根据上述数据和产品目录来决定。为了能够估算缺失值,我们剔除了包含超过 50%缺失值和出现不到 10 次的多值分类的分类属性。用这种方式得到的权值列在表 17.3 中。注意,表中没有提到的属性值为 0,因此,对 MDS 过程没有影响。根据我们的算法,品牌和内存大小是决定 MP3 播放器流行度最重要的属性,因此得到最高的权重。

表 17.3 对 MP3 播放器目录采用逐步泊松回归模型确定的权值。只显示了选中的属性

Attribute	Weight
Brand	0.242
Memory Size	0.176
Audio Formats	0.131
Battery Type	0.106
Width	0.090
Operating System	0.086
Color	0.084
Storage Type	0.084

在图 17.7 中,我们标出了两种图的产品点,图 17.7a 是 MSD 根据所有属性的权值都均等的方法得到的图,图 17.7b 是我们根据对应品牌赋权方法得到的图。第二种图通过 Procrustean 变化[16]旋转,与第一种图达到最佳匹配。由于品牌变得更重要,

同品牌的产品在第二个图中更容易被聚类。

图 17.8a 展示了第二种重要属性,即内存大小。该图展示的 MP3 播放器通常按左上角到右下角价格从低到高进行排序。最显著的属性——价格在表 17.3 中缺失,从而权值为 0。图 17.8b 是使用价格标记的图,图上有有一个明显的模式。因为价格和 MP3 的其他属性高度相关,如内存大小,因此这些属性能解释大部分物品的价格。

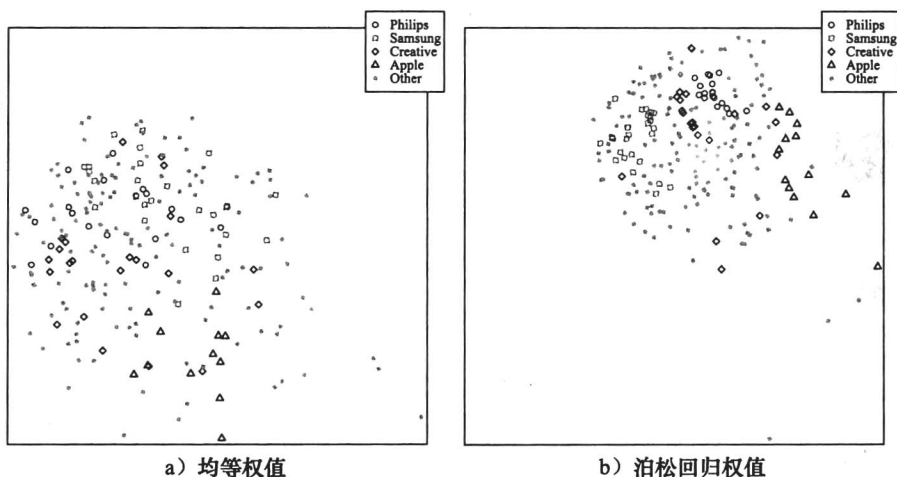


图 17.7 标示有品牌的基于 MDS 的产品目录图

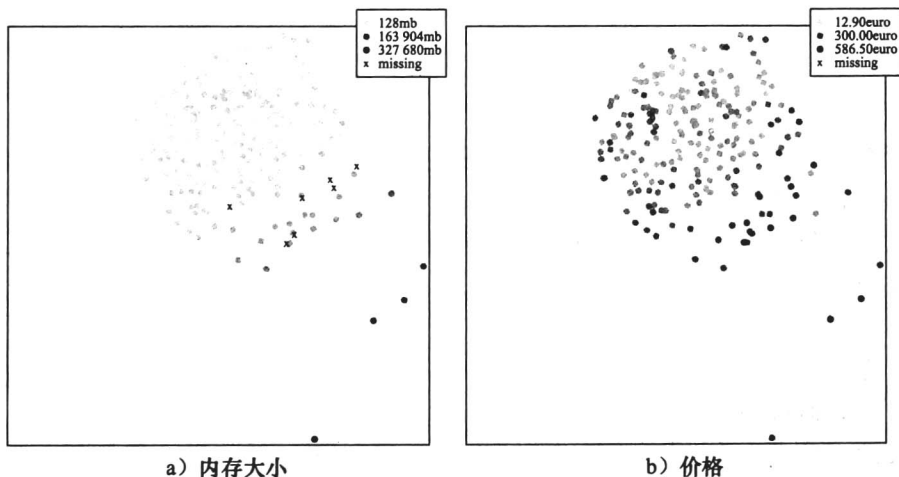


图 17.8 标示有内存大小和价格的基于 MDS 的产品目录图

更通用的是,我们可以发现 MDS 生成一种带离群值的圆形图,离群值和其他的产品非常不同。因此,图中大部分没被用到。另一方面,重要的属性显示出清晰的模式,图在第一眼就提供一个好的解释。

### 17.6.2 基于 NL-PCA 的产品目录图

图 17.9 显示了一个使用 NL-PCA 的原型截图。该应用可在 <http://www.browsingmap.com/mapvis.html> 获得。GUI 的主要部分是产品目录图。最开始,该图展示所有产品。少量(本例为 12 个)产品使用大的全彩图展示,其他产品使用较小的单色图。和基于 MDS 方法的产品目录图一样,对初始图使用 k-means 聚类过程来选择要突出显示的产品。

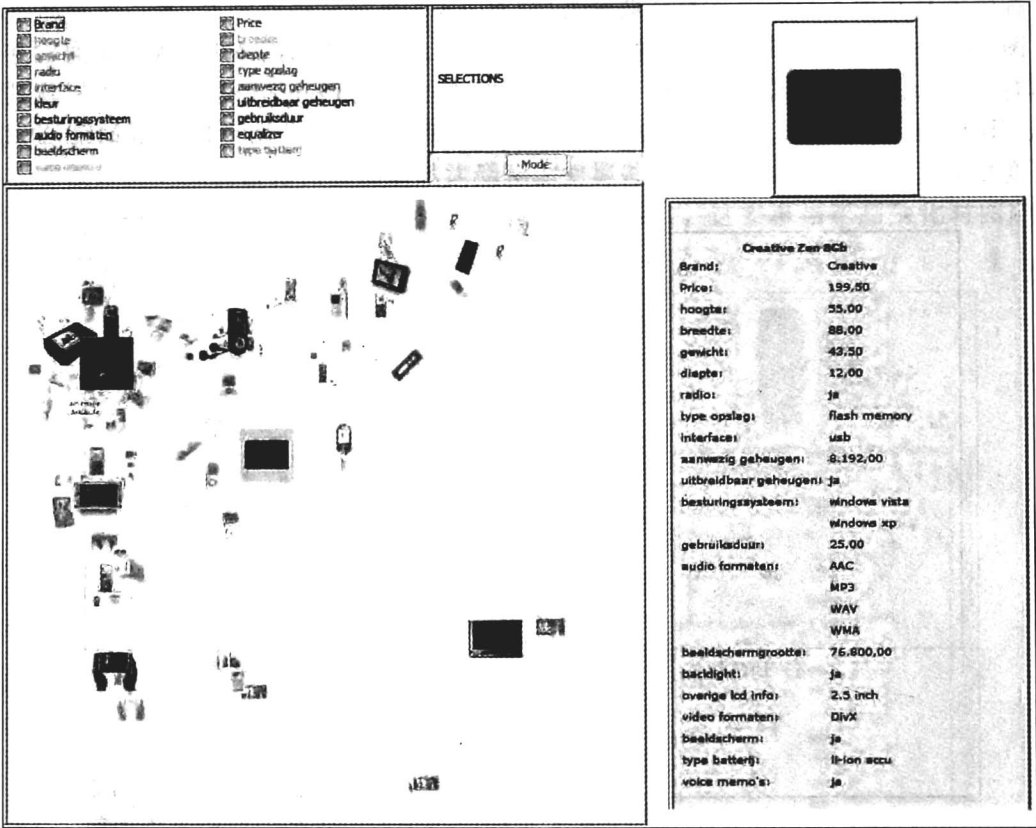


图 17.9 基于 NL-PCA 的产品目录图原型的截图

上述所有产品目录图都有一个产品属性列表，每个属性可以通过在选择框上点击来选择。如果一个属性被选择，这个分类的属性点将被增加到图上。该图所选属性的分类点不只是帮助用户解释图，同时也是导航整个图的工具。通过点击图上的分类点，目录被添加进选择列表中。

选择列表展示在属性列表的右边，决定了产品在图中高亮突出显示。被突出显示的产品集由如下方法决定。对每一个被选属性，一个展示产品应该属于至少一个被选分类。

当选择列表通过增加或移除分类点的方式调整后，图的可视化发生了改变。首先所有满足新约束的产品设为红色，其他为蓝色；然后从中随机选择一些产品高亮突出显示。

因为一次选择一般产生图的一个子集，也可能对部分图放大。然而，不能保证所有子空间节点满足选择列表中的全部约束。我们选择不删除这些产品点，因为它们可能是用户感兴趣的。虽然这些产品不满足用户的所有需求，但是它们和那些具有或者可能具有用户感兴趣属性的产品非常相似。

在完全和缩放图上，用户可以点击典型产品来得到一个完整的产品描述，显示在应用的右边。然而，在一个全彩图和单色图上滑动时，会显示一个包含该产品和重要属性值的提示框。另外，选择列表中的属性值也被展示在提示框中，当它们匹配偏好目录值时为绿色，反之为红色。由于 GUI 是基于单个 NL-PCA 图的，也可以像 MDS 产品目录图一样进行离线计算。

由于在含有大量缺失值的时候，NL-PCA 图的质量将变得非常差，我们剔除含超过 50% 缺失值的属性值。我们也剔除超过 50% 限制属性集中的产品。最后产生一个包含 19

个属性、189 个 MP3 播放器的集合，如表 17.2 所示。

在 NL-PCA 算法中，我们把所有数值属性当作有序的，有两个原因：第一，对用户来说很多数值属性没有线性解释，如内存大小；第二，由于潜在的单调回归过程，一些临界分类可以合并到一个单独的分类点。由于一个数值属性包含大量的分类（即数据集中的所有值都是唯一的），可视化所有的这些属性可能变得不清楚，并且使用这些分类值的选择可能没用。也因此，大量的分类只属于一个对象。使用一个序数转换会让这不是问题，因为少量物品会合并到它们的邻居。

图 17.10 展示了通过 NL-PCA 算法得到的两个双标图。每个双标图表示产品的属性值和产品分类值。并且，双标图中标记一个原点。根据 NL-PCA 方法的设计，普通常规产品（和其他很多产品属性相似的产品）应该更靠近原点，而独特产品应该远离原点，分类点同理。由于图 17.10 中的两种双标图都是由数值属性而来的，所以所有分类点在一条线上。像 MDS 图中一样，价格和内存大小彼此相关，并且在每个双标图的第二维上充分体现。

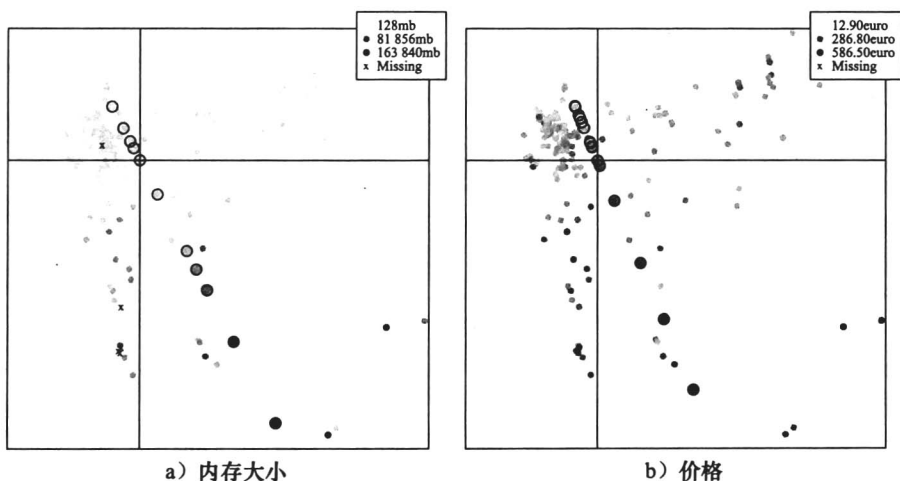


图 17.10 标有内存大小和价格的基于 NL-PCA 的双标图。大圈是分类点，小点代表目标点

第二个维度告诉我们关于内存大小和属性的相关性，而第一维区分新旧产品。如图 17.11 所示，我们使用界面属性标记产品。由于界面是一个多值分类属性，我们可以只通过单值标记每个产品，即产品目录中最频繁的一个。同时，我们只展示三个最常发生的分类点，所以所有产品至少属于他们中的一个。如上所述，这里有两个群，支持 USB 的旧 MP3 播放器和支持 USB 2.0 的新 MP3 播放器。对操作系统属性我们可以观察到类似的模式。

更普遍的，NL-PCA 方法创建的图比 MDS 方法使用了更多的可用空间。但是图的大部分是用于可视化特定而非一般的产品，而后者被聚类到图的右上角。同样，该图只展示不含过多缺失属性值的产品。

### 17.6.3 图像购物界面

图 17.12 展示一个图像购物界面原型。该原型可从 <http://www.browsingmap.com/mapvis.html> 获取。界面使用 4 个标号：实现图像购物界面的导航标签；实现图像推荐的直接搜索标签[23]；设置权值的偏好标签；保存产品的存储标签。

图像推荐系统根据一个搜索查询创建一个产品图，与该查询最相似的产品被呈现。使用图而不是通常的列表有一个优点，它能更清楚地知道哪些查询结果相似和当使用不同属性集的查询时会产生哪些变化。

在 GSI 和 GRS 图中，产品用缩略图显示，通过点击选择产品，并在图右边显示详细信息。在 GSI 标签的图上有进入下一次迭代、回到上一次迭代、重启和保存的按钮。

在偏好标签中用户可以设定 GRS 和 GSI 中计算相异度的权值。用户也可以取消某些属性，让它们的权值为 0。当权值变化时，图立即调整。该标签还有一个“智能权值”按钮，使用 17.4 节的方法决定权值，如表 17.3 所示。

GSI 两步之间有一个流畅转换。在用户选择某个产品后，新的产品被添加到图的随机位置。随后，图使用 MDS 进行优化。优化过程被展示给用户。当优化收敛时，旧产品逐渐变得越来越不重要（使用 MDS 版本的权重），直到它们的影响度为零。最终，旧产品从图中移除，新图达到最优。该实现产生流畅的可视转换，这对有效的 GUI 特别重要。

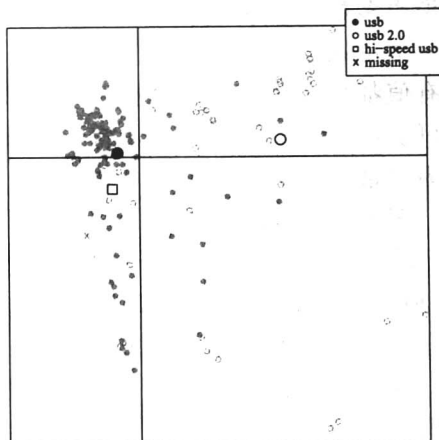


图 17.11 标有界面的基于 NL-PCA 的双标图。大块表明分类点，小的透明块代表目标点

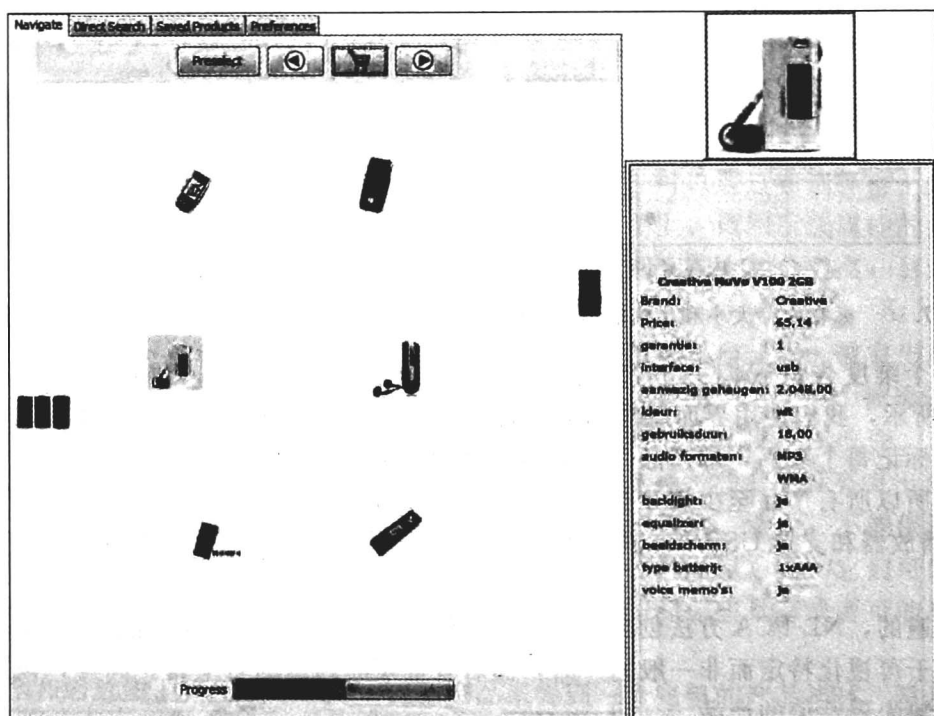


图 17.12 基于 MDS 的产品目录图原型的截图

当应用启动时，GSI 标签首先展示一个包含属性和属性值的树形结构，用来对下级产品的预选。通过这种方式，用户可以排除他明确不感兴趣的产品。

由于 GSI 图是动态生成的，依赖于用户输入和随机过程，我们无法在此给出结果。[23]使用了另一个 MP3 播放器目录，发现这些图理论上的质量（就压力值而言，参照



式(17.5))很高,主要是因为它们只显示 8 个产品。另外,推荐算法在仿真研究测试中也得到了理想的结果。

文献[23]中记录了一个基于 71 个调查对象的可用性研究。研究结果显示,虽然和传统基于列表的界面相比,用户觉得 GSI 更复杂,但仍没有明显的不满意。毕竟,这只是调查对象第一次使用 GSI。一些该可用性研究中对 GSI 不利的因素也应考虑进来。首先,在该实验中,由于图像推荐标签被关闭,用户错过了一种表达查询的方式。第二,用户错过了一种限制搜索空间的方式,这种方式是由一种新原型的选择树所提供的。

然而,GSI 方法的主要缺点似乎还是它的复杂性。基于该想法,静态的产品目录图可能是更好的选择。这些都还待进一步的可用性研究测试。

## 17.7 总结与展望

本章讨论了可以将产品目录可视化在一个图中,来提供一种使电子商务网站用户得到更好的可用产品的概览。在这样的图中,属性相似的产品应该被放得靠近彼此,而不相似的物品应该被放在图的不同区域。

在这本章的框架中,两种框架用于创建图:多维缩放(MDS)和非线性主成分分析(NL-PCA)。产品间的相似度尽可能地对应于二维空间中的距离。我们结合了 MDS 方法和 Gower 系数,使得它能灵活处理混合属性类和缺失值。在 MP3 播放器应用上使用的 MDS 方法似乎对产品簇和重要价格维度有一个清楚的解释。该图(以及 MDS)的主要缺点是有一个使得图打开的圆角,并且离群值位置相对图其他部分过远。但是,使用加强小相似度权值的方案可以克服该问题。

NL-PCA 是唯一能够在可视化产品旁可视化属性分类的方法。这些分类点可以用来选择图中的产品子集,就像我们原型中展现的那样。NL-PCA 图通常与 MDS 解释一致。虽然 NL-PCA 方法中离群产品占了图的大部分,但在图中更分散。NL-PCA 的主要缺点是,我们不能可视化所有产品,因为 NL-PCA 在物品包含太多缺失值时会生成很烂的图。另一个缺点是产品间的相似度不是像 MDS 方法那样直接对应于距离。虽然也可以通过不同的标准化方法完成,但分类点的解释就变得困难了,意味着不能再用于导航了。

由于用户通常不认为所有产品属性同等重要,我们展示了一种基于泊松回归模型的方法,可以通过点击流日志文件中的产品流行度自动决定属性的重要性权值。由于该方法独立于可视化技术,因此可用于涉及权值的技术,甚至可以用于涉及属性权值的推荐系统。该方法也有一个缺点,分类属性的权值是通过一种特别启发式的方式决定的,属性间的关联被忽略了。因此,我们用另一种方式决定这些权值,一种更灵活的基于加速回归树的模型。

通过引进图像购物界面,我们介绍了一种可结合推荐技术的基于图的可视化方法。当然,该方法也可结合其他基于内容和知识的推荐技术,如评价(参看文献[70]和第 13 章)。

除了和其他推荐方式的结合,我们认为产品目录可视化还存在着一些挑战。首先,由于决定图好坏是个人品味问题,和待可视化的数据有关,我们尝试用不同的可视化方法,如独立成分分析[8]或者投影追寻[12]。一个好主意是,在用户研究中比较不同的可视化方法。在该用户研究中,我们使用了一个合理大小的数据集。因为使用更大(成千上万)的产品目录,就意味着用于创建图的算法和界面本身也应该能够处理这样大的数。除了可视化海量产品目录类,另一个挑战是创建包含多类产品的图,如电子设备。

## 致谢

感谢 Compare Group 提供产品目录和点击流日志文件。

## 参考文献

1. Alhoniemi, E., Himberg, J., Parviainen, J., Vesanto, J.: SOM Toolbox 2.0 for Matlab 5. Laboratory of Computer and Information Science, Helsinki University of Technology, Helsinki, Finland (1999). Available at <http://www.cis.hut.fi/projects/somtoolbox/>
2. Bederson, B.B., Schneiderman, B., Wattenberg, M.: Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. *ACM Trans. Graph.* **21**(4), 833–854 (2002)
3. Borg, I., Groenen, P.J.F.: *Modern Multidimensional Scaling*, 2nd edn. Springer Series in Statistics. Springer, New York (2005)
4. Bruls, M., Huizing, K., Van Wijk, J.J.: Squarified treemaps. In: *Proceedings of Joint Eurographics and IEEE TCVG Symposium on Visualization*, pp. 33–42. IEEE Press (2000)
5. Chen, H., Houston, A.L., Sewell, R.R., Schatz, B.R.: Internet browsing and searching: User evaluations of category map and concept space techniques. *J. Am. Soc. Inf. Sci.* **49**(7), 582–603 (1998)
6. Chung, W.: Web searching in a multilingual world. *Commun. ACM* **51**(5), 32–40 (2008)
7. Chung, W., Bonillas, A., Lain, G., Xi, W., Chen, H.: Supporting non-English Web searching: An experiment on the Spanish business and the Arabic medical intelligence portals. *Decis. Support Syst.* **42**, 1697–1714 (2006)
8. Comon, P.: Independent component analysis, a new concept? *Signal Process.* **36**(3), 287–314 (1994)
9. De Leeuw, J.: Correctness of Kruskal's algorithms for monotone regression with ties. *Psychometrika* **42**(1), 141–144 (1977)
10. De Leeuw, J.: Convergence of the majorization method for multidimensional scaling. *J. Classif.* **5**, 163–180 (1988)
11. Donaldson, J.: Music recommendation mapping and interface based on structural network entropy. In: V. Oria, A. Elmagarmid, F. Lochovsky, Y. Saygin (eds.) *Proceedings of the 23rd International Conference on Data Engineering Workshops*, pp. 811–817. IEEE Computer Society (2007)
12. Friedman, J.H., Tukey, J.W.: A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. Comput.* **22**, 881–890 (1974)
13. Gifi, A.: *Nonlinear multivariate analysis*. Wiley, Chichester, UK (1990)
14. Gower, J.C.: A general coefficient of similarity and some of its properties. *Biometrics* **27**, 857–874 (1971)
15. Gower, J.C., Hand, D.J.: *Biplots, Monographs on Statistics and Applied Probability*, vol. 54. Chapman & Hall, London, UK (1996)
16. Green, B.F.: The orthogonal approximation of an oblique structure in factor analysis. *Psychometrika* **17**, 429–440 (1952)
17. Hartigan, J.A., Wong, M.A.: A k-means clustering algorithm. *Appl. Stat.* **28**, 100–108 (1979)
18. Hicklin, J.: Treemap for Matlab. Mathworks (2007). Available at <http://www.mathworks.com/matlabcentral/fileexchange/17192>
19. Honaker, J., King, G., Blackwell, M.: *Amelia II: A Program for Missing Data* (2008). R package version 1.1-27, available at <http://gking.harvard.edu/amelia>
20. Ibrahim, J.G., Chen, M.H., Lipsitz, S.R., Herring, A.H.: Missing-data methods for generalized linear models: A comparative review. *J. Am. Stat. Assoc.* **100**(469), 332–346 (2005)
21. Kagie, M., Van Wezel, M., Groenen, P.J.F.: Online shopping using a two dimensional product map. *Lect. Notes Comput. Sci.* **4655**, 89–98 (2007)
22. Kagie, M., Van Wezel, M., Groenen, P.J.F.: Choosing attribute weights for item dissimilarity using clickstream data with an application to a product catalog map. In: *Proceedings of the 2nd ACM Conference on Recommender Systems*, pp. 195–202. ACM Press, New York (2008)
23. Kagie, M., Van Wezel, M., Groenen, P.J.F.: A graphical shopping interface based on product attributes. *Decis. Support Syst.* **46**(1), 265–276 (2008)
24. Kagie, M., Van Wezel, M., Groenen, P.J.F.: An online shopping interface based on a joint product and attribute category map. In: *Proceedings of IUI Workshop on Recommendation and Collaboration ReColl 2008* (2008)
25. Kagie, M., Van Wezel, M., Groenen, P.J.F.: Determination of Attribute Weights for Recom-

- mender Systems Based on Product Popularity. Tech. rep. ERS-2009-022-MKT, Erasmus Research Institute in Management, Erasmus University Rotterdam (2009).
26. King, G., Honaker, J., Joseph, A., Scheve, K.: Analyzing incomplete political science data: An alternative algorithm for multiple imputation. *Am. Polit. Sci. Rev.* **95**(1), 49–69 (2001)
  27. Kohonen, T.: The self-organizing map. *Proc. IEEE* **78**(9), 1464–1480 (1990)
  28. Kohonen, T.: The self-organizing map. *Neurocomputing* **21**, 1–6 (1998)
  29. Kohonen, T.: *Self-Organizing Maps*, 3rd edn. Springer Series in Information Sciences. Springer, New York (2001)
  30. Kruskal, J.B.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* **29**(1), 1–27 (1964)
  31. Linting, M., Meulman, J.J., Groenen, P.J.F., Van der Kooij, A.J.: Nonlinear principal components analysis: Introduction and application. *Psychol. Methods* **12**(3), 336–358 (2007)
  32. McCullagh, P., Nelder, J.A.: *Generalized Linear Models, Monographs on Statistics and Applied Probability*, vol. 37, 2nd edn. Chapman & Hall, Boca Raton (1989)
  33. Meulman, J.J., Heiser, W.J.: SPSS Categories 15. SPSS Inc. (2007)
  34. Michailidis, G., De Leeuw, J.: The Gifi system of descriptive multivariate analysis. *Stat. Sci.* **13**(4), 307–336 (1998)
  35. Nelder, J.A., Wedderburn, R.W.M.: Generalized linear models. *J. R. Stat. Soc. Ser. A-Stat. Soc.* **135**(3), 370–384 (1972)
  36. Ong, T.H., Chen, H., Sung, W., Zhu, B.: Newsmap: a knowledge map for online news. *Decis. Support Syst.* **39**, 583–597 (2005)
  37. Pečenović, Z., Do, M.N., Vetterli, M., Pu, P.: Integrated browsing and searching of large image collections. *Lect. Notes in Comput. Sci.* **1929**, 173–206 (2000)
  38. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Tweaking critiquing. *Knowledge-Based Syst.* **18**(4–5), 143–151 (2005)
  39. Resnick, P., Varian, H.R.: Recommender systems. *Commun. ACM* **40**(3), 56–58 (1997)
  40. Ricci, F., Wöber, K., Zins, A.: Recommendation by collaborative browsing. In: A.J. Frew (ed.) *Information and Communication Technologies in Tourism 2005*, pp. 172–182. Springer, Vienna (2005)
  41. Rubin, D.B.: *Multiple Imputation for Nonresponse in Surveys*. Wiley, New York (1987)
  42. Sammon, J.W.: A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* **18**(5), 401–409 (1969)
  43. Schafer, J.L., Olsen, M.K.: Multiple imputation for multivariate missing-data problems: A data analyst's perspective. *Multivariate Behav. Res.* **33**(4), 545–571 (1998)
  44. Schneiderman, B.: Tree visualizations with tree-maps: 2-d space filling approach. *ACM Trans. Graph.* **11**(1), 92–99 (1992)
  45. Shimazu, H.: ExpertClerk: A conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artif. Intell. Rev.* **18**, 223–244 (2002)
  46. Stappers, P.J., Pasman, G., Groenen, P.J.F.: Exploring databases for taste or inspiration with interactive multi-dimensional scaling. In: *Proceedings IEA 2000 / HFES 2000, Ergonomics for the new Millennium*, pp. 3–575–3–578. Santa Monica CA (2000)
  47. Torgerson, W.S.: Multidimensional scaling: I. Theory and method. *Psychometrika* **17**, 401–419 (1952)
  48. Turetken, O., Sharda, R.: Developement of a fisheye-based information search processing aid (FISPA) for managing information overload in the web environment. *Decis. Support Syst.* **37**, 415–434 (2004)
  49. Van Gulik, R., Vignoli, F., Van der Wetering, H.: Mapping music in the palm of your hand, explore and discover your collection. In: *Proceedings of the 5th International Conference on Music Information Retrieval* (2004)
  50. Yang, C.C., Chen, H., Hong, K.: Visualization of large category map for Internet browsing. *Decis. Support Syst.* **35**, 89–102 (2003)



# 推荐系统与群体

第 18 章 个性化 Web 搜索中的群体、协作与推荐系统

第 19 章 社会化标签推荐系统

第 20 章 信任和推荐

第 21 章 组推荐系统



# 个性化 Web 搜索中的群体、协作与推荐系统

Barry Smyth、Mautice Coyle 和 Peter Briggs

**摘要** 搜索引擎是数以百万计的用户每天获取信息的主要手段；先进搜索引擎的成功与大规模使用是近十年来学术与工业进步的一个见证。然而，当前主流的搜索引擎仍然向他们的用户提供一刀切(one-size-fits-all)的服务，最终限制了结果列表(与用户查询)的相关性。本章将探讨在我们寻求新的、更加社会化的搜索引擎过程中，一些力求使 Web 搜索体验更个性化与协作的最新研究。

## 18.1 简介

Web 搜索引擎是当今最重要、最广泛使用的信息工具之一。主要的几个搜索引擎每月向数十亿用户提供搜索结果，同时，每年在这个过程中获得数十亿美元的广告收入。其中，谷歌是绝对的市场领导者，但这是它应得的，因为它解决了所有主要的网络搜索挑战，剩下的只是一些细微的算法改进。但现实是变幻莫测的，也许谷歌已经赢得了这一轮的搜索混战，但网络搜索战役还远未结束。

最近的一些研究表明，即使主流搜索引擎，如果给所有搜索关键词相同的用户返回相关的相关文档结果，其准确率也非常低。例如，在一个针对超过 20 000 个搜索查询的研究[24]中，研究人员发现：平均来看，谷歌返回的结果中，至少包含一个可用项的结果比例仅占 48%。也就是说，有 52% 的查询，搜索者不从返回结果中选择任何项。在这个问题上，很大程度上不仅是搜索引擎的问题，也是搜索者的问题：我们的查询往往是模糊而不确定的，很少可以明确地表达我们的搜索需求[100, 98, 99, 45, 90]。对于经常使用搜索引擎的人来说，我们已经习惯了这种成功率，并通过不断浏览更多搜索结果或者更改查询词来应对。但这充其量只能说明现在的搜索引擎还远没有达到它应有的高效——甚至有研究表明，信息工作者有 10% 的经费浪费在了搜索时间上[30]，最糟糕的是，很大一部分的搜索者根本无法找到他们需要的信息。

因此，尽管谷歌、雅虎以及其他公司不断地为成百上千万的用户提供强大的搜索服务，但是搜索仍然有很大的提升空间。本章将通过介绍一些最有可能革新已有技术、带来颠覆性创新的研究工作，来探究网络搜索的未来。我们会发现历史总是不断重演，正如过去的 10 年间，谷歌在网络搜索上的颠覆性革新为其带来的蓬勃发展一样，接下来的 10 年，新的搜索技术也将展示出其类似的对搜索市场的重整能力。

即使现在这种形式，现代搜索引擎也可以不严格地看作一种推荐系统：他们根据用户的查询，向用户返回一个推荐页面集。但是推荐技术在网络搜索中开始扮演越来越重要的

角色：从众多辅助搜索特征之一，慢慢地成为解决核心搜索挑战的技术之一。例如，现在的搜索引擎都添加了**查询推荐**服务作为核心搜索功能。当用户输入他们的查询时，类似谷歌推荐的服务利用推荐技术进行识别、排序并推荐之前已经查询成功的相关查询给用户，具体参见文献[81]。在本章中，我们将专注于网络搜索中的两类潜力巨大且非常强大的新思路——个性化和协同，你可以从最近的推荐系统研究[6, 53, 83, 35, 89, 77]以及本书第 5 章、第 4 章、第 13 章中找到他们的起源。他们质疑当前主流网络搜索的基本假设，并为传统网络搜索引擎提供改进建议。第一个基本假设主要关注当前主流网络搜索的一刀切特性——两个不同的用户输入相同的查询后，或多或少地，主流的搜索引擎都返回基本一致的结果列表，而置用户之间的不同偏好而不顾。他们认为网络搜索应该变得更个性化，以满足用户的隐含需求与偏好。我们将介绍多种不同的通过处理不同的用户偏好与上下文信息来提高搜索体验，实现网络搜索个性化的方法，实例参见[19, 23, 33, 97, 2, 48, 49, 108, 22, 69, 86, 14, 31]。第二个假设主要质疑网络搜索的孤立(solitary)特性。大部分的网络搜索都将搜索过程设计为单个用户与搜索引擎之间的交互。但是，最新的研究已经表明，很多情况下的信息搜索都具有明显的协同特征：成组的搜索者(如朋友、同事、同学)在搜索和分享结果时都通过多种方式合作。我们将介绍**协同信息检索**(collaborative information retrieval)领域的最新研究，该领域致力于发掘协同工作在各种信息发现任务中的潜能，实例参见[70, 71, 73, 72, 58, 59, 94, 1]。

另外，我们将重点介绍一种结合了个性化与协同特性的新的搜索服务：所谓的**社交搜索**服务。其将网络进化为一个可以促使用户在搜索过程中互动和协作的社交媒介，以便搜索者可以从其他想法类似的用户的偏好与经验中受益。的确，这种方法为搜索引擎的检索提供了新的信息来源：互动和协同信息。而且这些信息可以用于支持搜索中的推荐：基于词条的印证(term-overlap)与链接信息的原始搜索结果，可以通过基于搜索者偏好与活动的推荐结果得到很好的修订。这体现了推荐系统与搜索系统的结合趋势，就像连通关系的引入带来的搜索效果提升一样，我们有理由相信这些新的互动与偏好信息可以促使搜索引擎在“适当的时间为合适的用户提供合理的结果”这一目标上进入全新的发展阶段。

## 18.2 网络搜索历史简介

在介绍那些新兴的有潜力颠覆搜索行业的搜索技术之前，回顾一下网络搜索在过去 15 年间的发展还是很有必要的，这可以让我们更好地了解现代网络搜索的演进。最初的网络是没有搜索的。如果你想要访问某个特定的网页，你可以选择在你的浏览器中直接输入 URL，或者通过雅虎这样的导航网站作为入口，引导至你想浏览的页面。随着网络的不断成长、成长、再成长，简单的导航显然已经无法满足需求，因此，网络搜索开始出现，如 Lycos、Excite 和 Altavista 等都是早期的搜索引擎。

这些搜索引擎全都基于 20 世纪 70 年代便已经出现的称为**信息检索**(IR)的技术[104, 4]。图 18.1 展示了一个简化的典型搜索引擎原理图。简单来说，早期的搜索引擎通过以下方式对整个网络构建自己的索引：爬取网络上的页面并依次分析每个页面的内容，记录所包含的词组，以及它们的频率。当收到查询请求时，搜索引擎对包含查询词组的页面进行检索并排序，然后返回给用户。在网络搜索的初期，索引的大小是竞争的重点，索引了越多网络页面的搜索引擎比竞争对手具有明显的覆盖率优势。搜索结果的排序也会得到一些关注，但大多数情况下，这些搜索引擎都将查询词组在网页中的出现频率(相对

于将整个索引作为一个整体)作为相关性的决定性指标[96],因而更倾向于选择独特查询词出现频率大的页面。这种方式对于结构良好、封闭式的信息检索系统是合理的,这种系统中信息检索专家可以通过提交具体的、组织良好的查询获得合理的结果,但是当面对大规模的、具有异构特性的网络内容或者模糊的搜索查询时则表现糟糕。结果就是对于大部分的搜索用户,搜索体验都很糟糕:真正相关的结果被深藏在一堆可能看上去相似的结果列表中。

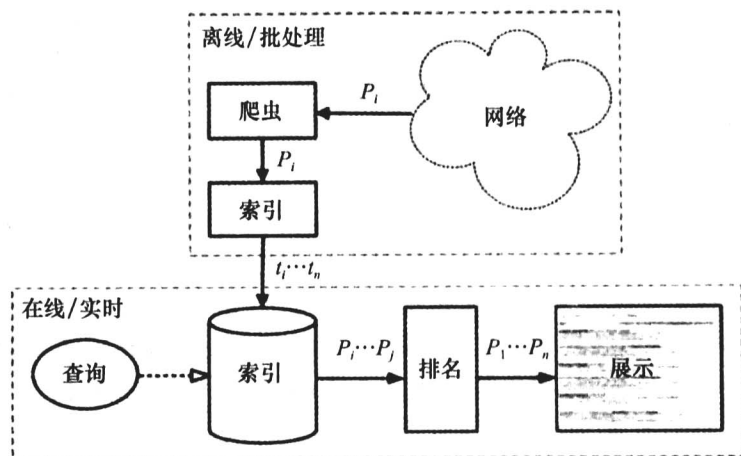


图 18.1 典型网络搜索引擎功能模块图。在离线处理过程中,一个页面,  $P_i$ , 通过爬虫定位到它的位置, 它的内容——一堆词组  $t_1, \dots, t_n$  被检索并构建索引。当收到一个搜索查询时, 引擎通过索引检索到匹配查询词的结果页面,  $p_1, \dots, p_j$ , 然后根据搜索引擎特定的排序方法对它们进行相关度排序, 组织成为结果列表返回给搜索用户

提高搜索结果的排序效果成为了当时这些早期搜索引擎的首要挑战。当面对这种迫切需求时, 即使构建最大规模索引的角逐也必须做出让步。但是, 人们很快认识到那些单纯依靠页面中的词条数的排序算法, 不论花费多少时间用于调整, 效果都是不明显的。简单地说, 当采用对查询与页面的词组进行计数来计算得分时, 很多的网页得分都是相当的, 但是仅有少数页面真正相关并且具有权威性。尽管词条匹配信息就整个相关性而言是有用的, 但是其本身的效果不显著。显然, 在排序过程中很多重要的信息被遗漏了。

这些遗漏的信息被 20 世纪 90 年代中期的多个研究组作为研究成果发表出来。包括 John Kleinberg[40]的工作, 以及最著名的、谷歌创始人 Larry Page 和 Sergey Brin[13]的工作。这些研究者最先开始利用网页之间的链接关系, 用这些信息评估网页之间的相对重要程度。Kleinberg、Page 和 Brin 将网络作为一种引文网络(示例见[60])。与一篇文章通过一个参考书目列表引用另一篇文章不同, 在网络上, 一个页面通过一个超链接引用另一个页面。更进一步, 直观上感觉, 一个页面的重要程度应该与那些链接到它的网页, 即所谓的反向链接呈某种函数关系。因此, 如果多个重要的页面链接到某个相同页面, 那么这个页面也应该被认为是重要的。这便是一种全新的评价某个页面重要程度的方法的基本出发点, 工作[40, 17]和[13]分别发明了全新的、用于从即使面对模糊的网络搜索查询, 也能识别出权威的、相关的页面的算法。到 20 世纪 90 年代后期, Page 和 Brin 的被称为 PageRank 的算法在第一版的谷歌中实现, 该算法结合了传统词组匹配技术与新的链接分析技术, 可以提供比当时其他搜索引擎结果更客观优秀的搜索结果。其余的, 正如他们说的一样, 已经成为历史。

### 18.3 网络搜索的未来

网络搜索无疑是一个重要的推荐挑战。网络的大规模与增长的特性,以及内容类型的巨大差异性,都带来了其特有的可怕的信息检索挑战。同时,随着网络用户结构的不断扩大,搜索引擎必须能够适应各种不同类型的用户以及用户的不同搜索水平。尤其对于大多数用户的检索水平,都达不到以文档为中心,基于词条的信息检索引擎对用户的期望,而这些引擎仍然是现代搜索技术的核心。这些引擎,以及他们所依赖的技术,很大程度上都假设查询是规整的、详细的,但实际网络搜索的查询远达不到这样的要求[36, 37, 100, 45]。事实上,大部分的网络搜索查询,跟搜索者真正的信息需求之间,往往是模糊和有歧义的,甚至查询中的某些词条在真正相关的目标文档(集)中根本就不出现。

由于大量的查询都得不到搜索者真正想要的结果,搜索引擎在这种基础性的搜索体验上还是有很大提升空间的。由于网络搜索查询的本质,上述问题会继续存在,至少有一部分会如此,因为指望用户的搜索技能在短时间内得到提升基本不大可能。因此,研究者为了提高整体的搜索体验,已经开始从两个方向进行探索研究。一个普遍的观点认为网络搜索应该变得更加个性化:用户的额外信息,如用户的偏好以及当时的上下文信息,应该用于构造更个性化的搜索方式,对搜索结果进行选取与排序,以得到更好的符合用户偏好和上下文的搜索结果(参见[86, 14, 31, 22, 2, 48])。另一种观点是网络搜索可以变得更协同:允许用户群组之间在搜索时进行(隐式或者显式的)合作(参见[70, 71, 73, 72, 58, 59, 94, 1])。

在接下来的章节中,我们会探究这些研究领域,介绍一些试图将静态的(非个性化的)、孤立的(非协作的)主流搜索引擎改造为更加个性化(18.3.1 节)或更加可协作(18.3.2 节)的搜索服务的开创性工作。这些开创性工作借鉴了推荐系统、用户属性、计算机支持下的协同工作研究(参见[84, 41, 89, 35, 52])。我们也将重点介绍一些力求将这两点同时结合以开创新一代个性化与可协作搜索服务的研究工作。在本章余下部分中,我们将这种混合搜索服务称为**社交搜索**,并对社交搜索的两种不同实现方式分别详细介绍两个案例研究。

#### 18.3.1 个性化网络搜索

许多推荐系统的设计目标就是向用户做出与他们特定环境或者个人偏好相吻合的推荐,比如,推荐系统帮助用户发掘如新闻文章[8, 9, 4]、书籍[46]、电影[54, 27, 42]甚至要买的产品[83, 74, 76, 51, 75, 20]等和个人相关的信息。而推荐技术在网络搜索中的应用可以(使搜索系统)摆脱主流搜索引擎中传统的一刀切方式。当我们想要实现个性化搜索体验时,有两个关键点:首先,我们需要理解搜索者的需求(用档案记录用户信息;然后,我们必须能够使用这些用户信息来影响搜索引擎的输出,比如,根据用户信息对搜索结果重排序,或者,影响网络搜索体验的其他方面。

为了更好地理解这些研究成果,我们有必要从两个重要的维度来考虑个性化网络搜索。一方面,我们可以从获得的用户信息的性质出发:一些方法专注于从用户当前的搜索上下文中获取短期用户信息(如[86, 14, 31]),而另外一些方法则适用于从一个时间段内用户的偏好来获取长期(long-term)的用户信息(如[22, 2, 48])。另一方面,可以从搜索阶段对这些用户信息的处理方式考虑:我们可以有效地区分那些以单个目标用户的特定用户信息为导向的方法(如[15, 89, 38, 43])和那些协同的方法,也就是说,他们以用户组的

用户信息为导向(如[46, 85, 41, 35, 90])。

通常来说,用户信息可以采用两种方式获得。**显式个性化**(explicit profiling)通过直接向用户询问不同类型的偏好信息:从类别偏好[22, 48]到简单的结果评分[2]。与之相反,**隐式个性化**(implicit profiling)技术更倾向于通过监控用户行为来推断偏好信息,而不干扰用户的搜索过程,如[22, 47, 69]。

通过显式个性化,用户自己完成个性化过程:或通过预先设置搜索偏好,或提供如对返回搜索结果进行评分等个性化相关反馈。Chirita 等[22]就使用了用户通过以下方式定义的独立用户信息:搜索者利用 ODP<sup>⊖</sup>网络目录分类体系,根据每个结果的 ODP 分类与用户信息之间的距离对搜索结果进行重新排序。他们试验了多种不同的距离度量方法后,公布了在线用户评估的新发现:他们的个性化方法可以提供比标准谷歌搜索更相关的结果排序。采用这种基于 ODP 分类方法的不足之一是仅有一小部分的网络内容是采用 ODP 分类的,返回的搜索结果中还有太多的内容因为没有这种分类信息而无法进行重排序。Ma 等[48]采用一种类似的方法,虽然他们的用户信息同样是显式地通过 ODP 类别信息进行表示的,但是他们对搜索结果的重排基于的是结果页面内容与用户个人 ODP 目录分类信息之间的余弦相似度。通过这种方式,搜索结果本身不再需要进行 ODP 分类。

与上述方法不同,ifWeb[2]通过一种对结构化要求更低的方式构建用户信息:他们将用户为了表达他们的特定信息需求而提供的关键字、自由文本描述,以及示例页面,保存为一个加权语义概念网络(weighted semantic network of concept)。ifWeb 同时还利用用户对结果的评分这种显式的相关性反馈来精炼和更新他们的。Wifs 系统[55]也采用了类似的方法,该系统从一堆词组中选择一些词组作为用户的用户信息初始值,并通过之后用户提供的查看过的文档信息来改善这些信息。采用这些显式方法的主要问题是大部分的用户都不愿意参与提供这些反馈[16]。此外,搜索者也许会发现很难对自己的信息需求进行分类,并一开始就提供精确的个人偏好。

另一种可能更成功的方式是隐式地推断用户的偏好(隐式个性化)。在工作[22]中,Liu 等[47]也使用 ODP 的层次类别表示用户信息,但是在这个工作中,这些类别信息是基于用户之前的搜索行为,比如,之前提交的查询以及选择的搜索结果文档,而自动获取的。很多算法已经被设计用于将这些搜索行为映射为 ODP 类别信息,包括基于最小二乘法线性拟合(Linear Least Squares Fit, LLSF)的方法[107],Rocchio 相关性反馈算法(Rocchio relevance feedback algorithm)[78],以及  $k$  近邻算法(kNN)[28]。在另一种与之相关的方法中,[103]使用统计语言模型,从这些长期的搜索历史中挖掘上下文信息,建立基于语言模型的用户信息。[69]也同样基于历史行为推断用户偏好,但是他们采用的是利用访问过的页面的浏览器缓存,来推断用户可能感兴趣的主观领域。这些主观领域或者类别信息,被合并入一个层次化用户信息中。在这个层次化用户信息中,每一种类型都根据用户在查看这个类别相关的页面时所花费的时间长短赋予权重。

以上的例子都是使用长期用户信息(long-term user profile)来试图捕获用户在一段时间内,至少在超过一个搜索会话的时间内的用户偏好。但是我们还可以选择短期用户信息,比如,一个典型的做法就是选择与当前搜索任务相关的上下文信息。举例来说,UCAIR 系统[86]利用最近提交的查询与选择的结果建立短期用户信息,用于个性化当前正在进行的搜索任务。在初始化一个新的搜索会话时,系统将为用户与他们当前的信息需



求建立新的用户信息。类似地, Watson[14]和 IntelliZap[31]都从当前的上下文信息建立了短期用户信息。Watson 通过识别用户正在编辑的本地文档或者正在浏览的页面中信息量大的词条,利用这些词组修改用户的搜索查询来获得个性化的结果。IntelliZap 的用户首先从他们正在浏览的文档中选择一个文本查询作为搜索初始值,然后在搜索系统的引导下,从该文档中选择相近的附加词对搜索查询进行扩展。在这些例子中,引导搜索结果个性化的用户信息记录的是与用户相关的、及时甚至临时的信息需求。

这些用户信息与(或)上下文信息的可用性是(搜索)个性化的前提,人们也已经开发了各种各样的技术来利用这些信息影响搜索体验的各个方面。这些技术不仅限于用来影响搜索结果的检索与排序,比如,事实上已经有人研究如何利用这些用户信息来影响整个网络搜索流程中的许多其他阶段,如对原始网页的抓取与索引[32, 44, 34, 29],以及查询生成[3, 7, 56]。比如,一个常用的基于用户信息个性化搜索结果的方式就是利用这些用户信息重写、细化或者扩展原始搜索查询,使得系统可以返回能更加体现搜索兴趣与上下文的特定结果。再如, Loutrika 和 Ioannidis[43]发明了一种他们称为 QDP(查询词消歧及个性化, Query Disambiguation and Personalization)的算法,用于根据词条间的加权关系(weighted relationships between terms)所表示的用户信息来扩展用户提交的查询。这些关系表现为词组间的各种操作关系:如连接(conjunction)、分裂(disjunction)、反义(negation)、置换(substitution),所以实际效果就是用户信息提供了一组个性化查询重写规则,用于在将提交上来的查询提交给搜索引擎之前对查询进行重写。Croft 等[26]描述了个性化语言模型可以用于用户信息,以提供查询扩展与相关反馈。从与个性化搜索高度相关的反馈立场来看,在用短期的、基于会话的用户信息去扩展查询和消除歧义的领域,也做了大量的研究工作[82]。从这个角度来说,这些方法与其说是针对个性化搜索本身,还不如说是在于提高搜索在一次独立搜索会话中的表现,而且其中的许多方法也可以用于指引更长期个性化搜索用户信息。

但是,利用用户信息进行个性化搜索最流行的方式可能还是直接影响搜索结果的排序。例如, Jeh 和 Widom[38]通过引入一个个性化的 PageRank[13],设定独立于搜索的先验条件,对网页的结果使用用户信息进行干预排序。这些用户信息由一些通过用户显式选取的、具有高 PageRank 值的偏好页面(preferred page)集构成,这些页面被用于为任意页面计算一个特定 PageRank 值:该页面与这些高分的偏好页面之间的相关性。Chirita 等[23]在这个思想基础上,通过分析搜索者的书签页面与历史网上冲浪行为自动选择这些偏好页面,利用一个 HubFinder 算法,找到适合驱动这种特定 PageRank 算法的相关高 PageRank 得分页面。这两种方法都基于从用户浏览历史中获得的长期用户信息。

Change 等[19]在 Kleinberg 的 HITS[39]排序算法基础上设计了一个个性化版本。他们的方法显式或隐式地从搜索者获得短期反馈,构建一个由个性化权威列表组成的用户信息,用户信息中的权威列表可以用于影响 HITS 算法以实现对搜索结果的个性化排序。使用了计算机科学研究论文作为语料库的实验结果显示,即使只有极少的搜索者反馈,个性化的 HITS 算法也能够显著改善符合搜索者偏好的结果排序。

另一种流行的基于排序的方法是,不调用搜索引擎的内部运行,仅利用用户的偏好对一些底层的(underlying)、通用的网络搜索引擎返回的结果进行重排序。Speretta 和 Guach[97]通过记录用户的查询以及在谷歌返回的结果中选择的的结果片段来建立个人,同时用户选择的结果片段被参照一个概念层次结构分类为一些概念的加权组合。之后所有来

自谷歌搜索的结果都根据每条结果与搜索者用户信息中的层次概念之间的相似度进行重新排序。Rohini 和 Varma[79]也提出了一种利用协同过滤技术对来自一个底层网络搜索引擎的结果进行重排序的个性化搜索方法,该方法利用的个性化用户信息采用隐式生成。

上述所有技术都专注于利用单一的用户信息(目标搜索者的偏好)来个性化用户的搜索体验。在推荐系统研究中,常用的方法是利用几组相关的用户信息为一个目标个体生成推荐。比如,有名的协同过滤(collaborative filtering)技术就是显式地利用一组与目标用户相似的用户的偏好来为目标用户生成推荐[77, 85, 46],参见[35, 52]以及第 21 章。类似的思想也已经开始影响网络搜索,事实上,我们将在 18.4 节介绍一种利用整个群体用户偏好的方法,虽然该方法在形式上用的整个群体的用户信息而不是一组独立用户的用户信息,具体参见[92, 90]。Sugiyama 等[101]提出了一种方法,所使用的长期用户信息就是通过相似用户生成的,这些相似用户则是利用改造的协同过滤算法根据用户的浏览历史而得到的。这个想法就是在过去关心了相似的查询且选择了相似结果的搜索者可以从他们共享的搜索偏好中受益。Sun 等[102]提出了一种类似的、基于协同过滤算法分析相关用户、查询与结果点击来个性化网络搜索结果的、名叫 CubeSVD 的方法。这两种方法都通过识别与当前搜索者相似的搜索用户来为单个用户建立更加综合性的用户信息。最新的研究[12]提出了一种点对点的方法来个性化网络搜索,采用的同样是在结果推荐中利用相似用户的用户信息。每个搜索者都根据之前的查询以及结果选择建立用户信息(同样还是长期用户信息)。在收到一个新的目标查询后,推荐仍然由用户自身的个人用户信息驱动,但是该查询同时也通过点对点搜索网络进行传播,使得连接的用户可以基于他们自己之前的搜索行为推荐相关结果。推荐结果根据与目标查询的相关性以及目标用户与相关节点的信任关系强度进行聚合和排序,读者可以阅读[63, 65, 64, 66, 67, 62]和第 20 章了解最近的基于信任的推荐技术。

### 18.3.2 协同信息检索

最近一些关于特定信息探索任务,如军事命令与控制任务或者医疗任务的研究已经有明显的证据表明,当信息在团队成员之间共享时,搜索类型的任务也可以协作[70, 71, 73, 72]。此外,[57]的研究更突出显示了更多通用网络搜索的固有协作属性。例如,在一项有超过 200 名调查对象的调研中,有明确迹象表明搜索行为出现了协作。超过 90% 的受访者表示,他们在搜索过程层面上频繁地参与协作。比如,87% 的受访者表示有过“背后搜索(back-seat searching)”行为,即他们站在搜索者的背后向搜索者提供替代查询建议。还有 30% 的受访者表示参与过搜索协调活动,即通过即时通信协调搜索。此外,96% 的用户表示有过搜索产出物层面的协作,即搜索的结果。比如,86% 的受访者在搜索过程中,通过电子邮件共享过自己找到的搜索结果。因此,尽管现在主流的搜索引擎没有明确的协作特征,但是有明确的证据显示用户在搜索中通过多种不同形式隐式地进行了协作,虽然根据[57]的报告,这些协作“变通”往往是令人沮丧和低效的。自然地,这促使研究人员考虑在未来版本的搜索引擎中支持不同类型的协作。

由此产生的协作信息检索方法可以从两个重要维度进行区分:时间——即同步(synchronous)还是异步(asynchronous)搜索——和位置——搜索者是本地的(co-located)还是远程的(remote)。本地系统为单个地点,最经典的就是一台 PC 的多个用户提供协作搜索体验(参见[1, 87]),而远程系统则允许搜索者在不同的地点使用多台设备进行搜索,参见[58, 59, 94]。前者显然可以通过利用本地搜索(co-located)具有的面对面本质特点,

大大地方便人员之间的直接协作,而后者则为协作搜索提供了更大的机会。另外,同步方法的特点是系统通过广播“呼叫搜索”(call to search),要求特定的参与者在一段精确设定的时间段内参与一个精确定义的搜索任务,示例参见[87]。与之相比,异步方法的特点是搜索任务更松散灵活,同时提供更加灵活的协作方式,使得不同的搜索者可以在一段时间内参与演变中的搜索会话,示例参见[58, 92]。

工作[1]给出了一个关于采用本地、同步方法实现协同网络搜索的很好的例子。他们的 CoSearch 系统设计用于提高那些计算资源有限的用户的搜索体验;比如,学校里同时使用一台 PC 的一组学生。CoSearch 是专门为利用周边可用的设备(如移动电话、闲置鼠标(extra mice)等)进行分布式控制与工作分解,同时保证团体的群体意识与沟通而设计的。例如,在一个一组用户通过仅有一台 PC,但有多个鼠标进行协作的场景中,CoSearch 支持一个主搜索者(a lead searcher)或者驱动者(可以操作键盘的人),其他用户则扮演搜索观察者(observers)。前者完成基本的搜索任务,但是所有用户可以使用返回的结果,独立地选择链接,将感兴趣的页面加入一个页面队列以供进一步回顾。CoSearch 接口同时还为用户提供多种方式对页面添加笔记。感兴趣的页面可以保存下来,同时用户在协作时可以通过 URL 以及对保存页面的笔记生成搜索总结。如果观察者可以使用手机,CoSearch 通过蓝牙连接为用户提供一系列扩展功能接口以支持丰富的独立功能。通过这种方式,观察者可以下载搜索结果到他们的手机,获取结果队列,添加页面到结果队列,以及与小组分享新页面。

CoSearch 的目的是演示在资源有限的环境中进行高效的网络搜索协作。其重点很大程度是把搜索劳动进行分工,但同时保持搜索者之间的交流沟通,而且在线用户调查也显示,CoSearch 在这方面是成功的[1]。文献[88]的工作在思想上与 CoSearch 相关,但侧重于使用桌面计算环境进行图像搜索的任务,这种情况非常适合本地用户共同搜索时的协作。初步的研究再次表明了这种方式在提高整体搜索的效率和协作上是有效的,至少在一些特定类型的信息获取任务上有效的,如图片搜索。文献[87]列举了一系列不同形式的同步搜索活动,在这些活动中,移动设备作为主要的搜索设备可以通过远程形式进行同步协作搜索。iBingo 系统允许一组用户通过每人使用一个 ipod touch 设备作为主要的搜索与反馈设备(尽管传统的 PC 似乎同样适用)在一个图片搜索任务中进行协作。有趣的是,CoSearch 的重点在很大程度是对搜索劳动的分工和对沟通的支持,而 iBingo 则使以下情况成为可能:即通过任何独立搜索者的相关反馈为其他用户提供好处。具体来说,iBingo 协作引擎使用活动中的每个用户的信息来鼓励其他用户探索不同的信息路径以及信息空间的不同方面。通过这种方式,用户正在进行的活动可以影响小组未来的搜索,在一定意义上,搜索过程就通过小组的搜索活动被“个性化”了。

远程搜索协作(异步或者同步)是 SearchTogether 的目标,它允许多组搜索者在搜索定位某个特定主题的信息时参与扩展共享搜索会话;参见文献[58]。简单地说,SearchTogether 系统允许用户参加共享搜索会话并邀请其他用户加入这些会话。每个用户可以独立地就一个特定主题搜索信息,但系统提供的功能可以允许个人用户通过推荐与评价具体的结果来与会话中的其他成员分享他们的发现。反过来说,SearchTogether 通过允许搜索者邀请其他人加入特定的搜索任务、允许合作搜索者通过多画面形式的结果接口同步查看每个搜索者的结果实现同步协作搜索。与上述的 CoSearch 一样,SearchTogether 的一个关键设计目标是支持对复杂、无限制的搜索任务进行分工。此外,该工作的另一个关键特征是其可以通过从接口级别降低搜索协作的开销实现在一个团队成员之间建立共享意识的功能。SearchTogether 通过包含一系列从消息集成、查询历史到最近查询推荐的功能,实现了上述目标。

总体来说,到目前为止,我们介绍过的所有协作信息检索系统都主要集中在支持劳动分工协作以及共享意识方面,并独立于底层的搜索过程。简而言之,这些系统都假定底层搜索可用,然后提供一个协作接口高效地直接引入搜索结果并允许用户分享这些结果。就像文献[68]中提到的,从被通知参与协作活动的搜索者都必须独立地检查和理解这些活动,以融合他们自己与合作搜索者的活动这个意义上来说,这些方法的一个主要局限就在于协作都仅限于系统提供的那些接口。因此,文献[68]的工作介绍了一种与底层搜索引擎资源联系更紧密的协作方法,使得协作搜索者通过不同方式参与的活动可以直接影响搜索引擎本身的操作。比如,中介技术(mediation techniques)用于优先选择至今还未出现的文档,而查询推荐技术则用于为进一步的搜索探索提供可选途径。

### 18.3.3 向社交搜索前进

到目前为止,我们分别集中讨论了以解决传统网络搜索核心问题为动机的网络搜索与信息发现领域的两个互补研究分支。研究者通过质疑主流网络搜索引擎一刀切的特性,提出了更加个性化的网络搜索技术;最近的研究通过质疑搜索主要是一个孤立的过程这一假设,突出展现了很多搜索场景的固有协作属性。

到目前为止,这些不同方向的研究根据动机和目标不同而分类。比如,在个性化搜索领域,大部分都是以产生更针对个人搜索者需求的结果列表为主导,而协作信息检索则集中于支持多组搜索者进行搜索劳动分工和提高合作搜索者之间的共享意识。但是,这两个研究团体都通过推荐系统领域这个共同点联系在一起,同时,推荐系统的视角为这两个不同分支的共同研究提供了可能。在接下来的内容中,我们将介绍两个试图将这两种技术结合以得到既个性化又可协作的搜索技术的相关案例研究。结果就是得到一个更加可协作——每个案例都假设有成组的搜索者参与——并且更个性化,尽管是在组这个层次而不是个人搜索者层次的网络搜索。这两个案例研究都将介绍远程、异步形式的协作网络搜索,同时将总结一些最近的在线用户调查的结果,来强调他们为终端用户带来的潜在好处。在每个例子中,我们都将介绍推荐技术在增加传统搜索结果列表价值上所起到的中心角色作用。例如,我们将介绍群体或者组成员的偏好与活动是如何用于促进在传统搜索结果列表中增加推荐搜索结果的。

## 18.4 案例研究 1: 基于群体的网络搜索

在第一个案例中,我们回顾最近基于群体的网络搜索工作。其中,有相似想法的用户的群体搜索活动被用于改善主流搜索引擎的结果,以提供更符合以群体为导向的结果列表;参看文献[91, 92]。它可能包括清晰界定或专门的群体。我们的目标是利用该群体搜索行为中查询的重复性和选择的规律性来作为推荐结果的来源。在本案例研究中,我们描述并评估该网络搜索方法的一种具体实现,该实现已用于主流搜索引擎,如 Google。

### 18.4.1 搜索群体中的重复性和规律性

在很多场景中,搜索可以视为群体推荐活动。例如,同一公司的员工可以充当一种有着重叠信息需求的搜索群体。类似地,同一班级的学生也可以作为一个搜索群体,因为他们都搜索与他们课堂作业相关的信息。同一主题网站的访客(如野外门户网站或驾车门户网站)将倾向于分享某些小兴趣并常使用网站的搜索工具来寻找相关信息。当然,社交网



络网站的朋友群也可以作为有共同兴趣的群体。

我们之所以对这些新兴的搜索群体感兴趣，是因为我们相信，这些群体成员的搜索模式之间在很大可能上存在着相似性。例如，图 18.2 展现了对当地软件公司 70 个员工最近 17 个星期搜索模式的研究结果。该研究先于后面讨论的试验。该研究中检查了超过 20 000 个人搜索查询和 16 000 个搜索选择。我们发现，平均下来，超过 65% 的查询词组同其他至少 5 个查询词组重合了至少 50% ( $>0.5$  相似度阈值) 的查询基本词；超过 90% 的查询词组同其他 25 个查询词组共享了至少 25% 的查询基本词。换句话说，该临时共同搜索群体的搜索者确实用相似方式搜索了相似的东西，远远超出典型一般搜索场景中出现的的大约 10% 的重复率(0.5 相似度阈值)[92]。

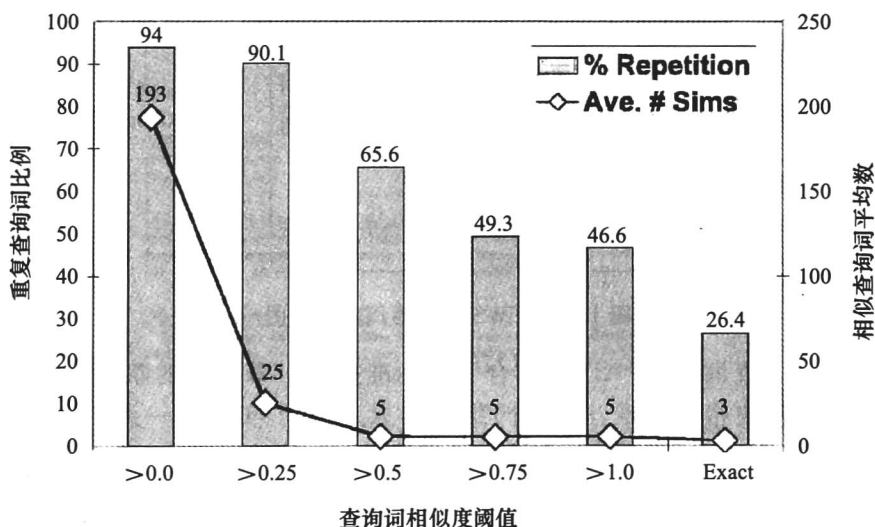


图 18.2 软件公司员工所用的搜索查询词的重复度和相似度

更多在其他搜索者群体上的研究证实了该重要结果[92]，从而激发了我们协同网络搜索的方法。该结果说明，在相似想法的搜索者群体的上下文中，网络搜索的世界是一个重复和规律的地方。在个人进行搜索的时候，他们的搜索经验生成了一种群体搜索知识。反过来，该搜索知识可以用于促进群体成员间的搜索经验分享。所以，作为一个简单的例子，当前面介绍野生门户网站的一个访客搜索“美洲虎图片”的时候，可以向其推荐使用过类似查询的其他群体成员所选中的结果。这些结果可能和该社团的野生兴趣相关，从而不需要任何花费很大代价对结果内容进行处理，我们就可以通过学习到的群体偏好来个性化搜索结果。通过这种方式，新搜索者可以分享到更有经验的搜索者的知识。

#### 18.4.2 协同网络搜索系统

图 18.3 展示了我们的协同网络搜索系统的基本架构，它用于辅助基础主流搜索引擎——本例为 Google。简单的想法是，采用基于代理的方法来拦截通往底层搜索引擎的查询，并处理从该引擎返回给用户的结果。通过这种方式，用户能在其他条件不变的情况下，得到他们感兴趣的搜索结果。但是在以协同网络搜索(CWS)的方式中，结果列表直接通过代理得到提升。例如，用户  $U_i$  提交给 Google 一个查询  $q_T$ ，该请求重定向给 CWS 系统，于是两件事发生了：首先，该查询传递给 Google，并以正常方式返回一个结果列表  $R_S$ 。其次，该查询同时也用于访问  $U_i$  的群体搜索活动的一个本地存储——CWS 命中矩



阵，来生成一个排序的提升候选集  $R_p$ ，如下所述。解释引擎通过展示群体历史的图形化表示来对该提升候选项进行解释。结果列表  $R_p$  和  $R_s$  合并后的最终列表  $R_{final}$  被返回给用户。通常这种合并会将最相关的  $k$  个(如  $k=3$ )改进项提升到结果列表的前列。

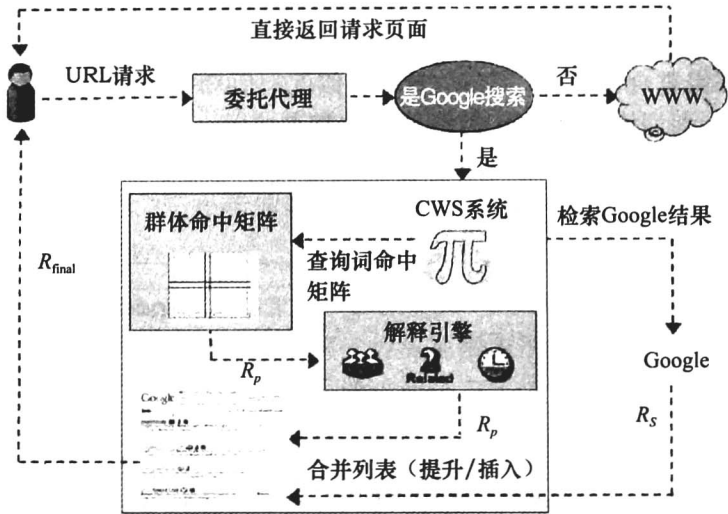


图 18.3 CWS 系统的代理结构图

因此对目标搜索查询来说，CWS 结合了由标准搜索引擎得到的默认结果列表  $R_s$ ，以及由群体过去搜索历史得到的推荐(提升)结果集  $R_p$ 。为了实现这一点，给定群体的用户搜索历史  $C(C=\{U_1 \dots, U_n\})$  存放于命中矩阵  $H^C$  中，其中每一行对应于查询  $q_i$ ，每一列对应于结果页  $p_j$ ， $H_{ij}^C$  的值表示群体  $C$  的成员通过  $q_i$  选择  $p_j$  的次数。用这种方式，每个命中矩阵作为一个群体搜索经验的知识库和群体成员查询相关的结果。

$$\text{Relevance}(p_j, q_i) = \frac{H_{ij}}{\sum_j H_{ij}} \tag{18.1}$$

$$\text{Sim}(q, q') = \frac{|q \cap q'|}{|q \cup q'|} \tag{18.2}$$

$$\text{WRel}(p_j, q_T, q_1, \dots, q_n) = \frac{\sum_{i=1 \dots n} \text{Relevance}(p_j, q_i) \cdot \text{Sim}(q_T, q_i)}{\sum_{i=1 \dots n} \text{Exists}(p_j, q_i) \cdot \text{Sim}(q_T, q_i)} \tag{18.3}$$

当响应一个新目标查询  $q_T$  时， $H_C$  用于识别和排序过去常选择的结果。查询  $q_i$  和结果  $p_i$  的相关性可以通过以往  $q_i$  对应的  $p_i$  的相对频率来估算，如式 (18.1) 所示。更一般的情形是，我们可以把  $q_T$  相似的查询的结果放入可供选择的结果池(见式 (18.2))，并通过相关性的加权模型进行排序，如式 18.3 所示，它通过关联查询的相似性来衡量每个结果的相关性。注意当查询  $q_i$  有对应的结果页  $p_j$  时，谓语 Exist 返回 1，反之返回 0。

图 18.4 和图 18.5 展示了谷歌对查

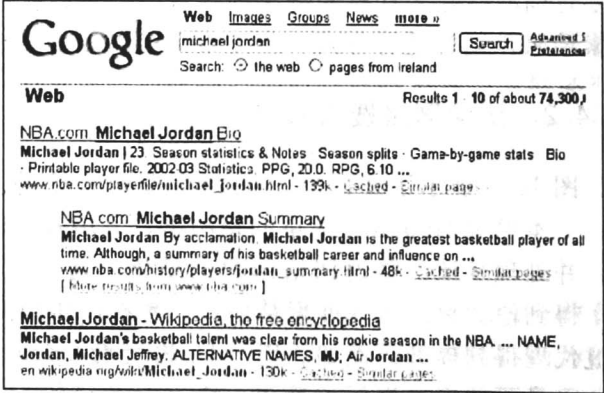


图 18.4 在 Google 中搜索 Michael Jordan 的结果页

询“Michael Jordan”返回的结果列表的屏幕快照。在图 18.4 的例子中，我们可以看到默认的谷歌结果列表，篮球运动员明显占据了整个结果。然而在图 18.5 中，我们可以看到，结果列表被 CWS 的基于代理的版本所修改，(本例)通过一个计算机科学研究者群体训练。结果通过标准谷歌接口所展示，我们发现，最前面 3 个结果成了著名的伯克利大学教授<sup>①</sup>。另外，提升的结果通过解释图标进行注解，该图标用于捕获结果群体历史的不同方向，包括捕获群体成员结果流行性，即最近被选择的频繁度的信息和其他产生该选择的查询的信息。

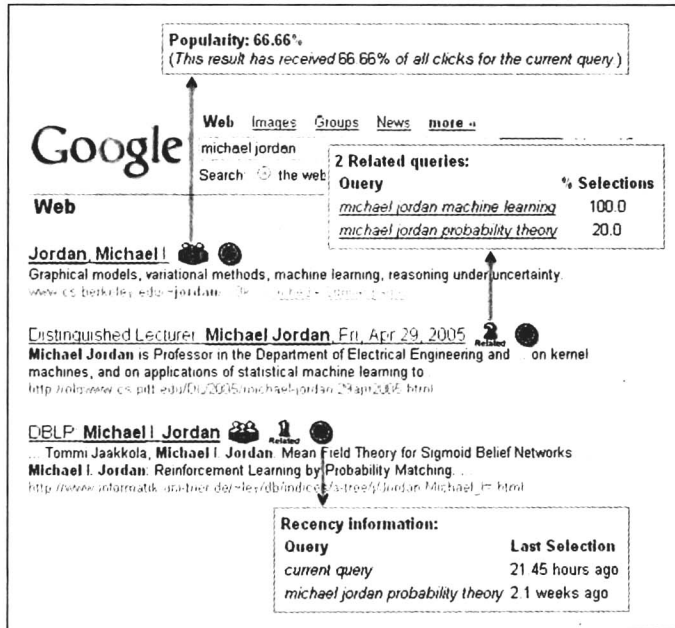


图 18.5 由计算机科学研究者群体所训练出来的基于 CWS 系统所返回的 Michael Jordan 搜索结果页。其中当鼠标滑过后会展现额外的信息

### 18.4.3 评估

目前基于代理的架构用来作为协同搜索场景中 CWS 方法的长期试验基础。本节将描述该试验的最新结果，它将证明 CWS 提供的基于群体提升的价值。

该试验参与者包括一个本地软件公司 Dublin 的 70 多位员工。CWS 架构与标准 Google 搜索引擎配置在一起共同运行，以便所有的 Google 请求被重定向给 CWS 系统。搜索体验是基于标准谷歌接口，对任意会话允许最大 3 个结果提升(以及解释)。如果超过 3 个可用的提升，那么没有提升的结果使用解释图标进行标注，但仍留在默认的谷歌位置中。此处展示的结果来自超过 10 周的使用数据和覆盖了总计 12 621 的个人搜索会话。

评价自然环境中新搜索技术的一个挑战是，如何评价个人搜索会话的质量。理想情况是，我们捕获用户搜索时直接的相关反应。但是要求用户在每个会话中或者选择特定结果时提供这样的反馈，相对突兀。该方法在当前的实验中是不可行的，因为参与者想要确保他们的搜索经验不会偏离常规，因此不愿意接受弹出窗口、填表或者其他形式的额外反馈。作为一种替代，在本章评价中，我们使用较为间接的相关性，它基于成功会话的概念(参看[92, 91])。我们定义一个成功的会话为，至少一个搜索结果被选择，表明用户发现

<sup>①</sup> Michael Jordan 是机器学习方向的泰斗之一。——译者注

至少一个(部分)相关结果。反之,一个用户没有选择任何结果的搜索会话则被认为是失败的,意味着用户没有发现任何相关结果。虽然这是对整体搜索质量的一种相对粗糙的衡量,但它至少允许我们系统地比较搜索会话。

关于带提升的会话(promoted session)和不带提升的标准会话(standard session)的成功率比较展示在图 18.6 中。该结果显示,在 10 周的实验过程中,带提升的会话平均要比只包含谷歌结果的标准会话更有可能成功。这几乎 30% 的收益是由于基于群体的提升。换句话说,在实验过程中,我们发现,超过一半的使用标准 Google 搜索会话的用户找不到任何值得选择的结果。相反,在同一时期,同样的搜索者在包含了群体提升的会话中,有着明显更大的成功率,只有不到 40% 的会话不能成功地吸引用户做出选择。

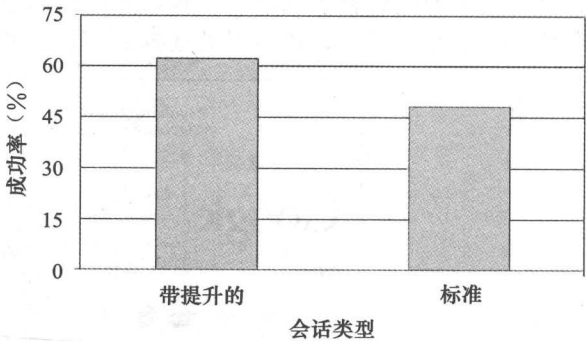


图 18.6 提升前后的会话成功率对比

在企业中,当涉及整体搜索生产力时,这些结果会有重要的影响。因为在许多知识密集的商业场景中,可以通过剔除的失败搜索会话得到很大的提升。例如,在国际数据团体(IDC)的一个最新报告[30]中发现,知识工作者平均花费 25% 的时间用于搜索信息,从而一个有着 1000 个知识工作者的企业每年将会因为不能定位和检索信息,浪费掉 250 万美元(有时达到 1500 万美元)。在该情形下,任何显著减少失败搜索会话的方法都对提升企业生产力有着巨大的作用,对更大型的组织尤其如此。

18.4.4 讨论

这里的协同网络搜索模型力求利用自然出现的搜索重复性和群体相似想法的搜索者间结果选择的规律性。在本例研究中,我们关注一个特别的搜索群体,员工群。明显地,这是一个相当简单明确的群体,不奇怪我们能在协同网络搜索中发现其高度重复性和规律性。尽管如此,由执行相似信息查找任务的个人所聚集起来的群体非常普遍,不管是同一公司的员工,或是同一班级的学生,或是同一研究组的研究员,都属于该群体类。

当然还有很多其他群体类。例如,我们已经提到的场景,对同一主题网站的访客组成的群组可以视为一个特别搜索群体。更普遍地,有趣的是人们将群体挖掘和识别看作开放式的问题,目前也有很多研究致力于采用各种方法自动识别在线群体,具体可以参看[11, 5, 21, 106, 105]。为了更好地理解新社会网络世界中的在线群体的本质,最好提供一种更灵活的搜索协作形式,并通过更灵活和动态地定义搜索群体定义得以促进。

18.5 案例研究 2: 网络搜索共享

之前的案例站在协同网络搜索的群体视角,有相似想法的用户群体的搜索活动用于影响主流搜索引擎结果。本节描述另一种协同网络搜索模型。该模型在 HeyStaks 系统中实现,在方式上两点重要的不同:首先,HeyStaks 对协同网络搜索采用了更多的用户导向方法,帮助用户更好地组织和分享他们的搜索体验。为了做到这一点,HeyStaks 允许用户创建和分享搜索体验的知识库,而不是协调在搜索群体中的参与度。其次,我们对搜

索引引擎整合采用了非常不一样的方式。我们没有采用上一研究案例中描述的基于代理的方法，而是把一个浏览器工具栏整合进主流搜索引擎，如 Google 中。该浏览器工具栏能让协同搜索引擎具有捕获和引导搜索活动的能力。最后，我们将总结最近关于真实用户研究的发现，来探索 HeyStaks 用户群中出现的搜索协同本质。

### 18.5.1 HeyStaks 系统

HeyStaks 在主流搜索引擎中加入了两个基本特征。第一，它允许用户创建搜索堆 (search stak)——一种在搜索时记录搜索体验的文件夹。用户可以把搜索堆与其他用户分享，让他们把他们的搜索也添加进来。第二，HeyStaks 使用用户的搜索堆来生成推荐，并将其加入主流搜索引擎的结果中。这些推荐是之前的搜索堆的成员认为对类似查询相关的结果，帮助搜索者发掘他的好友或同事觉得有意思的结果，否则这些结果很可能永远埋在 Google 的默认结果列表中。

如图 18.7 所示，HeyStaks 采用两种形式的基本组件：一个客户端的浏览器工具栏和一个后端服务器。工具栏允许用户创建和分享搜索堆，并提供一些如打标签和给网页投票等辅助功能。工具栏也捕获搜索点击和管理 HeyStaks 推荐和默认结果列表的整合。后端服务器管理个人堆索引 (个人网页与查询/标签关键词和正面/负面投票的索引)、堆数据库 (堆标题、成员、描述、状态等)、HeyStaks 社会网路服务以及推荐引擎。在余下内容中，我们将简单概述 HeyStaks 的基本操作，然后关注推荐引擎背后的细节。

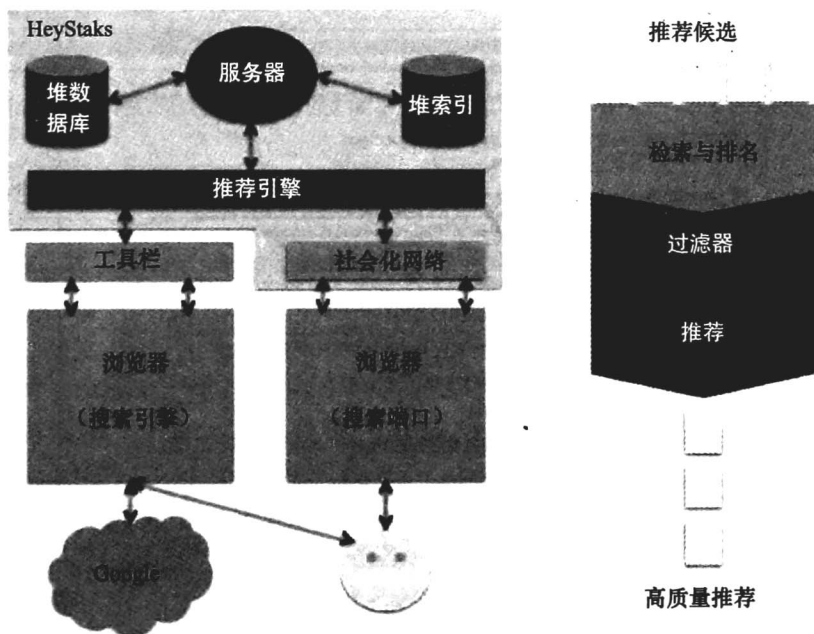


图 18.7 HeyStack 系统架构图以及离线推荐模型

考虑一个生动的例子。Steve、Bill 和一些朋友计划一次欧洲旅行，他们将使用网络搜索作为主要的信息获取源，来了解该做什么和该访问哪些景点。Steve 创建了一个叫作“欧洲旅行 2008”的 (私有) 搜索堆，并把它分享给了 Bill 和其他朋友，鼓励他们在旅游相关的搜索中使用该搜索堆。

图 18.8 中，Steve 在开始新的关于“都柏林旅馆”的搜索时选择了该搜索堆，结果展示在图 18.9 中。在 Google 的结果中，HeyStaks 额外增加了两个推荐，因为最近“欧洲旅行

2008”搜索堆的其他成员通过选择、投票、标签的方式，发现这些结果相关。这些结果可能存在于 Google 结果中很深的位置或者甚至不在该查询的结果集中。其他相关结果也将被 HeyStak 强调，但是仍留在 Google 结果中的原位。通过该方式，Steve 和 Bill 从他们以往的相似搜索中彼此受益。此外，HeyStaks 可以根据其他相关的公共搜索堆进行推荐，帮助用户从其他群组或群体创建的搜索知识中获益。

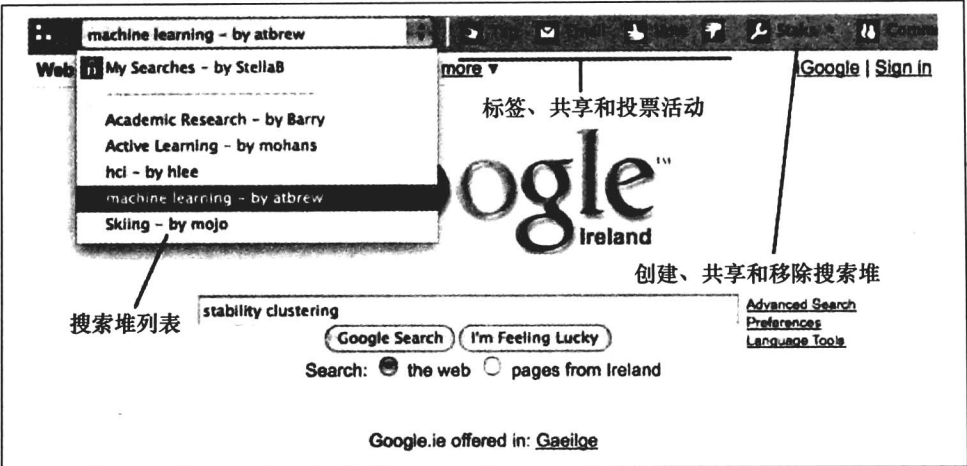


图 18.8 选择一个新的搜索堆

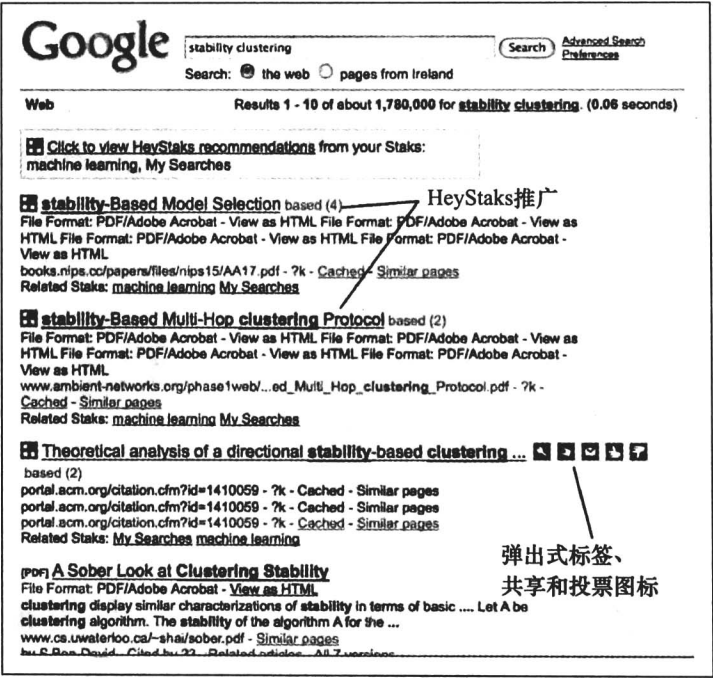


图 18.9 基于 HeyStack 的 Google 搜索结果

独立于工具栏，Heystaks 的用户也可以从 HeyStaks 搜索门户中获益，该门户提供一个用户搜索历史的社会网络。例如，图 18.10 显示了“欧洲旅行 2008”搜索堆的门户主页，它对所有的堆成员开放。它展示一种最近搜索历史和查询云活动的反馈，以使用户找出其他人搜索过的东西。该搜索门户也提供给用户更广的功能，例如，搜索堆维护(在搜索堆之内和之间进行编辑、移动、复制操作)，丰富的搜索和过滤工具，以及管理他们自己搜



索文档和新搜索伙伴的大量特征。

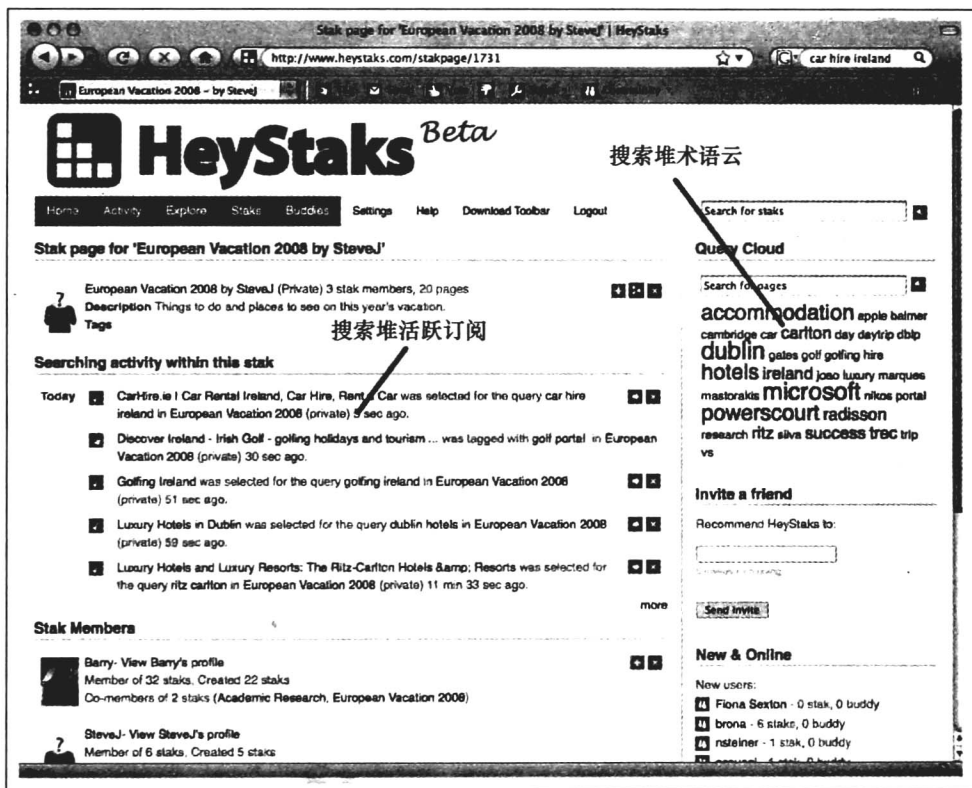


图 18.10 HeyStack 搜索门户提供了搜索堆和历史搜索记录的直接访问入口

### 18.5.2 HeyStaks 推荐引擎

在 HeyStaks 中，每一个搜索任务(S)作为一个该搜索堆成员搜索活动的文档。HeyStaks 结合了大量隐式或显式的文档来捕获丰富的搜索经验历史。每个任务由一个结果集( $S = \{p_1, \dots, p_k\}$ )组成，并且每个页面匿名地和大量隐式和显式的指示器关联，包括结果被选择(sel)的总次数、产生该结果的查询关键词( $q_1, \dots, q_n$ )、结果被打上标签的次数和标签关键词( $t_1, \dots, t_m$ )、接受的投票( $v^+, v^-$ )，以及分享过的人数(share)(显式的兴趣反馈)，如式(18.4)。

$$p_i^S = \{q_1, \dots, q_n, t_1, \dots, t_m, v^+, v^-, sel, tag, share\} \quad (18.4)$$

通过该方式，每个页面与一个词条数据(查询词条或标签词条)和一个使用行为数据(选择、标签、分享、投票统计)的集合关联起来。词条数据以 Lucene(lucene.apache.org)索引表表示，其中每个页面与相关联的查询和标签关键词索引起来，提供检索和排序提升候选的基础。使用行为数据提供额外的置信源，用于过滤结果并生成最终推荐列表。搜索时，推荐结果通过许多阶段生成：相关结果从 Lucene 堆索引中检索并排序；这些推荐候选通过置信模型剔除噪声推荐；剩余的结果通过一系列推荐规则添加到 Google 结果列表中。

简单地说，有两种有价值候选集：首要价值是那些从活跃搜索堆  $S_i$  中产生的；次要价值是从搜索者搜索堆列表中产生的。为了生成这些有价值的候选，HeyStaks 服务器使用当前查询  $q_i$  作为每个搜索堆索引  $S_i$  的探针，来识别相关搜索堆页的  $P(S_i, q_i)$  集合。每个候选页面  $p$  使用 Lucene 的 TF-IDF 检索函数式(18.5)进行打分，作为初始化推荐排序的基础。

$$\text{score}(q_i, p) = \sum_{t \in q_i} \text{tf}(t \in p) \cdot \text{idf}(t)^2 \quad (18.5)$$

搜索堆不可避免地会有噪声，因为他们常常会包含那些主题外的页面。例如，搜索者常常忘记在搜索会话开始时设置一个合适的搜索堆，虽然 HeyStaks 包含了许多自动的搜索堆选择技术来确保正确的搜索堆被激活，但这些技术仍不完美，误分类不可避免地会发生；参看文献[18, 95]。从而，检索和排序的阶段可能选择了和当前查询内容不严格相关的页面。为了避免产生假推荐，HeyStaks 采用了置信过滤，使用一系列阈值模型，根据特定结果的使用记录，来估算其相关性；标签使用记录被认为比投票更重要，而后者又比隐式使用记录更重要。例如，只被一个搜索堆用户选择过一次的页面，自动地不被考虑进来，其他类似的情况也同样进行剔除。该模型的详细细节超出了本书的范围，但进一步考虑，能够肯定的是，任何不满足必要使用记录阈值的结果都将被剔除。

在使用记录修剪后我们得到了改进的首要 and 次要推荐，最后的任务就是将这些合格推荐加入 Google 推荐列表中。HeyStaks 使用若干不同的推荐规则来决定怎样和在哪添加这些推荐。同样地，篇幅的限制使得我们无法对该组件进行详细的解释，但是，像前 3 个主提升项总是被加入到 Google 结果列表的前面，并使用 HeyStaks 的推荐图标进行标记。如果一个余留的首要推荐仍然在默认的 Google 结果列表中，那么它将被标记在合适的位置。如果仍有余留首要推荐，那么它们将会加入次要推荐列表中。次要推荐列表通过 TF-IDF 分数进行排序，然后作为一个可选的可扩展的推荐列表加入 Google 的结果列表中；更多详情请参看文献[93, 94]。

### 18.5.3 评估

本节测试了早期使用了工具栏和服务 beta 版本以来仍活跃的部分 95 位 HeyStaks 用户。他们在 2008 年 10~12 月间进行注册，下面的结果是他们 2008 年 8 月到 2009 年 1 月的使用总结。我们的目的是更深地理解用户如何使用 HeyStaks 以及是否从搜索推荐中获益。因为这是一个在非实验环境下对真实用户的研究，所以对测量有一些必然的限制。比如，没有实验对照组，而且主要出于数据隐私的原因，通过对比 Google 默认结果和 HeyStaks 推荐结果来分析用户的相关点击行为也是不可行的。但是对于感兴趣的读者来说，我们之前的工作记录了对该分析的更常规的对照组实验研究[10, 25, 92]。

HeyStaks 主张的关键是搜索者需要用更好的方式来组织和共享他们的搜索体验。HeyStaks 提供这些特征，但这些用户是否真的肯花时间来创建搜索堆？他们是否愿意与他人共享或者加入其他用户的搜索堆？

在 HeyStaks 初始部署的过程中，用户确实创造了可观的搜索堆并进行了分享。例如，根据图 18.11，测试版用户平均创建超过 3.2 个新搜索堆并参加超过 1.4 个。也许这不奇怪：大部分用户创建和分享少量搜索堆给小圈子内的同事和朋友，至少在刚开始的时候。

总计有超过 300 个搜索堆创建在大范围的主题上，从宽泛的主题如旅游、研究、音乐和电影，到小兴趣包括建筑、黑白照片和山地车。有少数用户是大量搜索堆的创建者和参与者：一个用户创建了 13 个搜索堆并加入了另外 11 个搜索堆，同其他 47 个研究者(共享相同的搜索堆的用户)建立了研究网。事实上平均下来，每个用户通过共享的搜索堆仅经过 5 个其他研究者就能连接到研究网。

多数的搜索堆是公共的，虽然大部分(52%)只含一个成员，即创建者。因此，48%的搜索堆分享给了其他至少 1 个用户，事实上这个数字平均下来是 3.6。另一种看法如

图 18.11b 所示：70% 的用户分享和参加了搜索堆，只有 30% 的用户只为自己创建搜索堆并拒绝加入其他用户的搜索堆。

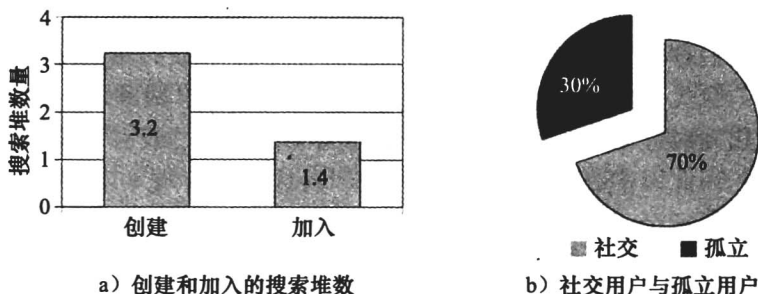


图 18.11 每个用户的平均搜索堆创建数和加入数以及社交用户和孤立用户的比率

HeyStak 的核心思想是受这样的观点启发：网络搜索是天然的社会化或协同活动。尽管主流搜索引擎没有进行支持，研究者也确有寻找替代的协同频道(如 email、IM 等)来分享他们的搜索体验，尽管这并不高效，参看示例[57]。HeyStak 早期一个最重要的问题是，在何种程度上他们的自然搜索活动能有助于创建协同搜索者群体。像用户搜索、标签和投票一样，他们有效地生产和消费群体搜索知识。一个用户可能是第一个选择或打标签某个结果作为一个搜索堆，在这种情况下，他们生产了新的搜索知识。接着，如果该结果被推荐给另一个用户并重选择(打标签或投票)，那么该用户消费了该搜索知识。当然，他们也生产了搜索知识，因为他们的选择、标签或投票将加入搜索堆中。

我们发现 85% 的用户参与到了搜索协同中。其中大多数消费了至少一个其他用户生产的结果，平均下来，这些用户消费了 7.45 个其他用户的结果。作为对比，50% 的用户生产了至少被一个其他用户消费的知识，本例中的生产者创建的知识平均被超过 12 个其他用户消费。

我们应该问一个问题，在何种程度上，个人用户倾向于成为搜索知识的生产者或消费者？是否一些搜索者是搜索知识的网络生产者，即他们更倾向于创建对他人有用的搜索知识？是否另一些用户是网络消费者，即他们更倾向于消费他人创建的搜索知识？该数据如图 18.12a 所示。为了弄清楚这一点，网络生产者定义为：帮助过的人比帮助自己人更多的用户；而网络消费者定义为帮助自己人比帮助过的人更多的用户。图 18.12 显示 47% 的用户是网络生产者。值得注意的是，我们在上面提到有 50% 的用户生成了被其他用户消费过的搜索知识。这也意味着绝大部分的用户，事实上是 94% 的用户，帮助的人比帮助自己的人更多。

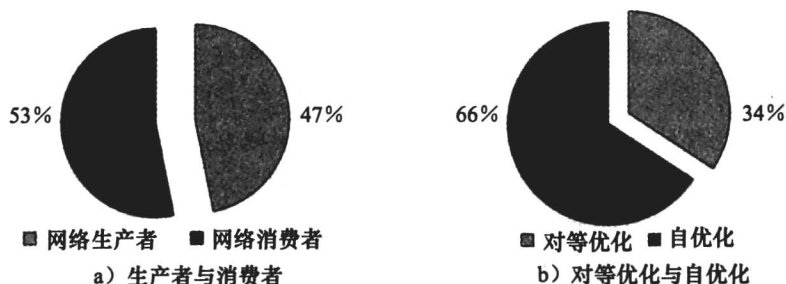


图 18.12 网络生产者和消费者的比率以及对等优化和自优化的比率

所以，我们发现大量用户帮助别人，同时大量用户被人帮助。也许，这种利他主义只

限于小数量的搜索?也许,用户在个人搜索的大多时候靠的还是自己?换个视角看上述分析可以帮助解释该问题,即观察用户在搜索时判断足够相关而选择的优化项的根源。总的来说,测试版的用户在搜索时选择了超过 11 000 个优化项。一些优化项是从搜索者自己的历史记录来的,我们称为自优化;其他的则是从分享了搜索堆的用户来的,我们称为对等优化。直观上讲,自优化的选择对应于 HeyStaks 中帮助用户回顾他们之前找到的结果的例子,而对等优化则对应于发掘任务,帮助用户关注可能错过或者很难找到的新内容;具体参看[61, 50]。因此图 18.12b 比较对等优化和自优化,发现 66% 的优化选择来自于搜索者自己的历史搜索记录,大多时候 HeyStaks 帮助用户重新获得以前发现的结果。然而,对等优化被选择的比例为 34%(我们已经知道它们来自很多不同的用户),可以帮助搜索者挖掘他人发现的新信息。

对自优化的偏差并不奇怪,特别是考虑搜索者的习惯和搜索堆开发的早期阶段。大多数搜索堆早期的增长都是通过单个用户,通常是创建者,然后大部分的页面优先就生成了——作为创建者自己的搜索查询的回应。大部分的优化是自优化的某些页面,即来自领导者自己的搜索活动。很多搜索堆并没有分享,因此只能用于生成自优化。随着搜索堆的分享和更多用户的加入,搜索者堆变得更多样。更多的结果被对等用户的动作所加入,更多的对等优先被生成和选择。在之后的工作中,一个有意思的任务是,探索搜索堆的演化,来研究越来越多的用户是如何影响搜索堆内容和优化的。目前比较满意的是,即使在搜索堆演化的早期,每个搜索堆平均也能有 3~4 个成员,在 34% 的时间里,成员受益于对等用户动作所产生的优化。

#### 18.5.4 讨论

同第一个案例研究比较,HeyStak 促进了一种更直接形式的搜索协同——搜索堆显式地被用户创建和分享——最后微搜索群体形成了,研究者小组可以基于某个主题或题目进行协作。当然不排除大协同小组的形成,而且很有可能某种搜索堆演化成搜索群体的方式,与之前案例研究所预期的非常吻合。

本案例研究遗留了许多未解问题,因此未来仍有很大的研究空间。例如,潜在的搜索堆增长将产生全新的推荐机会,因为用户将会从加入什么搜索堆的推荐中受益。此外,考虑合并或分解搜索堆也将会有意思的事情,例如,用户可以结合已有的搜索堆创建新的搜索堆。

### 18.6 总结

网络搜索引擎是并且毫无疑问将继续是我们用于探索在线信息的主流工具。对所有如 Google 的主流搜索引擎的成功来说,网络搜索问题仍然远远没有解决,对新一代网络搜索技术的研究也在完善中。在未来很有可能主流搜索引擎将会演化到为用户提供更多的支持,使用户在对的时间找到对的信息,而推荐技术注定会扮演重要的角色。

例如,研究者已经知道如何通过结合用户个人资料和推荐技术,让搜索引擎对我们特别的、个人的需求和偏好做出更多的响应,以呈现更个性化的用户体验,不管是通过目标结果列表的生成还是改进的查询推荐。另一个研究则试图通过提供给搜索者培养和改进小组甚至大的搜索者群体的新工具,来寻求利用很多网站搜索任务的内在的协同本质。

本章通过考察若干代表性的系统和技术,提供了自主研究中一些有趣领域的简单介

绍。接着强调了这些互补的协同个性化网络搜索方法是如何结合起来作为协同辅助，来为用户提供更好的个性化的，推荐系统在其中扮演了一种新的社交搜索服务中心角色。就这方面我们提出了两种不同的社交搜索系统案例研究，来展现在实际中如 Google 的主流搜索引擎是如何通过这些方法进行改进的。

在未来，随着推荐系统在网络搜索中扮演着日益重要的角色，主流搜索引擎很有可能采纳这些方法中的原理而进化。今天网络搜索很大地依赖于个人搜索者，我们相信推荐技术的引入将会使搜索引擎变得更有前瞻性，因为他们将会参与而不是响应用户的信息需求。同时，这会提供很多研究机会，特别是在搜索接口的层面，因为我们要寻找能使推荐技术和网络特性相结合的新方法。的确，我们已经看到一些早期的例子如谷歌、雅虎将查询推荐技术结合到常规搜索框中。但是，这只是一个开始，当研究者面临性能、隐私等挑战时，搜索引擎将为下一代推荐技术提供一个独立的平台。正如早期作为推荐系统平台的电子商务网站一样，搜索引擎将会为推荐系统时代赢得更多追随者。

## 致谢

该工作得到 Science Foundation Ireland 的授权 07/CE/I1147。

## 参考文献

1. Saleema Amershi and Meredith Ringel Morris. Cosearch: a system for co-located collaborative web search. In *Proceedings of the annual SIGCHI conference on Human factors in computing systems (CHI)*, pages 1647–1656, 2008.
2. Fabio A. Asnicar and Carlo Tasso. Ifweb: a prototype of user model-based intelligent agent for document filtering and navigation in the world wide web. In *Proceedings of the Workshop on Adaptive Systems and User Modeling on the World Wide Web at the Sixth International Conference on User Modeling*, pages 3–11, 1997.
3. Ricardo A. Baeza-Yates, Carlos A. Hurtado, and Marcelo Mendoza. Query recommendation using query logs in search engines. In *Current Trends in Database Technology - EDBT 2004 Workshops*, pages 588–596, 2004.
4. Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.
5. Lori Baker-Eveleth, Suprateek Sarker, and Daniel M. Eveleth. Formation of an online community of practice: An inductive study unearthing key elements. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 254–256, 2005.
6. M. Balabanovic and Y. Shoham. FAB: Content-Based Collaborative Recommender. *Communications of the ACM*, 40(3):66–72, 1997.
7. Evelyn Balfe and Barry Smyth. Improving web search through collaborative query recommendation. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 268–272, 2004.
8. D. Billsus and M. Pazzani. A Hybrid User Model for News Story Classification. In *Proceedings of the Seventh International Conference on User Modeling, UM '99*, 1999. Banff, Canada.
9. Daniel Billsus, Michael J. Pazzani, and James Chen. A learning agent for wireless news access. In *IUI '00: Proceedings of the 5th international conference on Intelligent user interfaces*, pages 33–36, New York, NY, USA, 2000. ACM Press.
10. Oisín Boydell and Barry Smyth. Enhancing case-based, collaborative web search. In *Proceedings of International Conference on Case-Based Reasoning (ICCBR)*, pages 329–343, 2007.
11. John G. Breslin, Andreas Harth, Uldis Bojars, and Stefan Decker. Towards semantically-interlinked online communities. In *European Semantic Web Conference (ESWC)*, pages 500–514, 2005.
12. Peter Briggs and Barry Smyth. Provenance, trust, and sharing in peer-to-peer case-based web search. In *Proceedings of European Conference on Case-Based Reasoning (ECCBR)*, pages 89–103, 2008.



13. Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107-117, 1998.
14. Jay Budzik and Kristian J. Hammond. User interactions with everyday applications as context for just-in-time information access. In *IUI '00: Proceedings of the 5th international conference on Intelligent user interfaces*, pages 44-51, New York, NY, USA, 2000. ACM.
15. R. Burke. The Wasabi Personal Shopper: A Case-Based Recommender System. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*. AAAI Press, 1999.
16. John M. Carroll and Mary Beth Rosson. Paradox of the active user. In John M. Carroll, editor, *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*, chapter 5, pages 80-111. Bradford Books/MIT Press, 1987.
17. Soumen Chakrabarti, Byron Dom, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins, David Gibson, and Jon M. Kleinberg. Mining the web's link structure. *IEEE Computer*, 32(8):60-67, 1999.
18. Pierre-Antoine Champin, Peter Briggs, Maurice Coyle, and Barry Smyth. Coping with noisy search experiences. In *Twenty-ninth SGAI International Conference on Artificial Intelligence (AI-2009)*. Springer-Verlag, 2009.
19. Huan Chang, David Cohn, and Andrew McCallum. Learning to create customized authority lists. In *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning*, pages 127-134, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.
20. Li Chen and Pearl Pu. Evaluating critiquing-based recommender agents. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2006.
21. Alvin Chin and Mark H. Chignell. Identifying active subgroups in online communities. In *CASCON*, pages 280-283, 2007.
22. Paul Alexandru Chirita, Wolfgang Nejdl, Raluca Paiu, and Christian Kohlschütter. Using odp metadata to personalize search. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 178-185, New York, NY, USA, 2005. ACM Press.
23. Paul-Alexandru Chirita, Daniel Olmedilla, and Wolfgang Nejdl. Pros: A personalized ranking platform for web search. In Paul De Bra and Wolfgang Nejdl, editors, *Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH)*, volume 3137 of *Lecture Notes in Computer Science*, pages 34-43. Springer, 2004.
24. Maurice Coyle and Barry Smyth. Information recovery and discovery in collaborative web search. In *Proceedings of the European Conference on Information retrieval (ECIR)*, pages 356-367, 2007.
25. Maurice Coyle and Barry Smyth. Supporting intelligent web search. *ACM Trans. Internet Techn.*, 7(4), 2007.
26. W. Bruce Croft, Stephen Cronen-Townsend, and Victor Larvrenko. Relevance feedback and personalization: A language modeling perspective. In *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, 2001.
27. B.J. Dahlen, J.A. Konstan, J.L. Herlocker, N. Good, A. Borchers, and J. Riedl. Jump-starting movieLens: User benefits of starting a collaborative filtering system with "dead-data". In . University of Minnesota TR 98-017, 1998.
28. B. V. Dasarathy. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, Los Alamitos, CA, 1991.
29. R. Dolin, D. Agrawal, A. El Abbadi, and L. Dillon. Pharos: a scalable distributed architecture for locating heterogeneous information sources. In *CIKM '97: Proceedings of the sixth international conference on Information and knowledge management*, pages 348-355, New York, NY, USA, 1997. ACM.
30. Susan Feldman and Chris Sherman. The High Cost of Not Finding Information. In *(IDC White Paper)*. IDC Group, 2000.
31. L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppín. Placing search in context: the concept revisited. In *WWW '01: Proceedings of the 10th International Conference on the World Wide Web*, pages 406-414. ACM Press, 2001.
32. C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. Citeseer: an automatic citation indexing system. In *DL '98: Proceedings of the third ACM conference on Digital libraries*, pages 89-98, New York, NY, USA, 1998. ACM.
33. Laura A. Granka, Thorsten Joachims, and Geri Gay. Eye-tracking analysis of user behavior in www search. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 478-479, New York, NY, USA, 2004. ACM.

34. Luis Gravano, Héctor García-Molina, and Anthony Tomasic. Gloss: text-source discovery over the internet. *ACM Trans. Database Syst.*, 24(2):229–264, 1999.
35. Anthony Jameson and Barry Smyth. Recommendation to groups. In Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors, *The Adaptive Web*, pages 596–627. Springer-Verlag, 2007.
36. Bernard J. Jansen and Amanda Spink. An analysis of web searching by european alltheweb.com users. *Inf. Process. Manage.*, 41(2):361–381, 2005.
37. Bernard J. Jansen, Amanda Spink, Judy Bateman, and Tefko Saracevic. Real life information retrieval: a study of user queries on the web. *SIGIR Forum*, 32(1):5–17, 1998.
38. Glen Jeh and Jennifer Widom. Scaling personalized web search. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 271–279, New York, NY, USA, 2003. ACM.
39. Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
40. Jon M. Kleinberg. Hubs, authorities, and communities. *ACM Comput. Surv.*, 31(4):5, 1999.
41. J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gorgan, and J. Riedl. GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87, 1997.
42. Yehuda Koren. Tutorial on recent progress in collaborative filtering. In *Proceedings of the International Conference on Recommender Systems (RecSys)*, pages 333–334, 2008.
43. G. Koutrika and Y. Ioannidis. A unified user-profile framework for query disambiguation and personalization. In *Proc. of the Workshop on New Technologies for Personalized Information Access*, pages 44–53, 2005.
44. A. Kruger, C. L. Giles, F. M. Coetzee, E. Glover, G. W. Flake, S. Lawrence, and C. Omlin. Deadliner: building a new niche search engine. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 272–281, New York, NY, USA, 2000. ACM.
45. S. Lawrence and C. Lee Giles. Accessibility of Information on the Web. *Nature*, 400(6740):107–109, 1999.
46. Greg Linden, Brent Smith, and Jeremy York. Industry report: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Distributed Systems Online*, 4(1), 2003.
47. Fang Liu, Clement Yu, and Weiye Meng. Personalized web search by mapping user queries to categories. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 558–565, New York, NY, USA, 2002. ACM Press.
48. Zhongming Ma, Gautam Pant, and Olivia R. Liu Sheng. Interest-based personalized search. *ACM Trans. Inf. Syst.*, 25(1):5, 2007.
49. Christos Makris, Yannis Panagis, Evangelos Sakkopoulos, and Athanasios Tsakalidis. Category ranking for personalized search. *Data Knowl. Eng.*, 60(1):109–125, 2007.
50. Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.
51. Kevin McCarthy, James Reilly, Lorraine McGinty, and Barry Smyth. Experiments in dynamic critiquing. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*, pages 175–182, 2005.
52. Kevin McCarthy, Maria Salamó, Lorcan Coyle, Lorraine McGinty, Barry Smyth, and Paddy Nixon. Cats: A synchronous approach to collaborative group recommendation. In *Proceedings of the International FLAIRS Conference*, pages 86–91, 2006.
53. L. McGinty and B. Smyth. Comparison-Based Recommendation. In Susan Craw, editor, *Proceedings of the Sixth European Conference on Case-Based Reasoning (ECCBR 2002)*, pages 575–589. Springer, 2002. Aberdeen, Scotland.
54. Ryan J. Meuth, Paul Robinette, and Donald C. Wunsch. Computational intelligence meets the netflix prize. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 686–691, 2008.
55. Alessandro Micarelli and Filippo Sciarra. Anatomy and empirical evaluation of an adaptive web-based information filtering system. *User Modeling and User-Adapted Interaction*, 14(2-3):159–200, 2004.
56. M. Mitra, A. Singhal, and C. Buckley. Improving Automatic Query Expansion. In *Proceedings of ACM SIGIR*, pages 206–214. ACM Press, 1998.
57. Meredith Ringel Morris. A survey of collaborative web search practices. In *Proceedings of the annual SIGCHI conference on Human factors in computing systems (CHI)*, pages 1657–1660, 2008.
58. Meredith Ringel Morris and Eric Horvitz. S<sup>3</sup>: Storable, shareable search. In *INTERACT (1)*,

- pages 120–123, 2007.
59. Meredith Ringel Morris and Eric Horvitz. Searchtogether: an interface for collaborative web search. In *UIST*, pages 3–12, 2007.
60. Sridhar P. Nerur, Riyaz Sikora, George Mangalaraj, and Venugopal Balijepally. Assessing the relative influence of journals in a citation network. *Commun. ACM*, 48(11):71–74, 2005.
61. Vicki L. O'Day and Robin Jeffries. Orienteering in an information landscape: how information seekers get from here to there. In *CHI '93: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 438–445, New York, NY, USA, 1993. ACM Press.
62. John O'Donovan, Vesile Evrim, and Barry Smyth. Personalizing trust in online auctions. In *Proceedings of the European Starting AI Researcher Symposium (STAIRS)*, Trento, Italy, 2006.
63. John O'Donovan and Barry Smyth. Eliciting trust values from recommendation errors. In *Proceedings of the International FLAIRS Conference*, pages 289–294, 2005.
64. John O'Donovan and Barry Smyth. Trust in recommender systems. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*, pages 167–174, 2005.
65. John O'Donovan and Barry Smyth. Trust no one: Evaluating trust-based filtering for recommenders. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1663–1665, 2005.
66. John O'Donovan and Barry Smyth. Is trust robust?: an analysis of trust-based recommendation. In *Intelligent User Interfaces*, pages 101–108, 2006.
67. John O'Donovan and Barry Smyth. Mining trust values from recommendation errors. *International Journal on Artificial Intelligence Tools*, 15(6):945–962, 2006.
68. Jeremy Pickens, Gene Golovchinsky, Chirag Shah, Pernilla Qvarfordt, and Maribeth Back. Algorithmic mediation for collaborative exploratory search. In *SIGIR*, pages 315–322, 2008.
69. Alexander Pretschner and Susan Gauch. Ontology based personalized search. In *ICTAI '99: Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence*, page 391, Washington, DC, USA, 1999. IEEE Computer Society.
70. Madhu C. Reddy and Paul Dourish. A finger on the pulse: temporal rhythms and information seeking in medical work. In *CSCW*, pages 344–353, 2002.
71. Madhu C. Reddy, Paul Dourish, and Wanda Pratt. Coordinating heterogeneous work: Information and representation in medical care. In *ECSCW*, pages 239–258, 2001.
72. Madhu C. Reddy and Bernard J. Jansen. A model for understanding collaborative information behavior in context: A study of two healthcare teams. *Inf. Process. Manage.*, 44(1):256–273, 2008.
73. Madhu C. Reddy and Patricia Ruma Spence. Collaborative information seeking: A field study of a multidisciplinary patient care team. *Inf. Process. Manage.*, 44(1):242–255, 2008.
74. J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Dynamic Critiquing. In Peter Funk and Pedro A. Gonzalez Calero, editors, *Proceedings of the 7th European Conference on Case-Based Reasoning*, pages 763–777. Springer-Verlag, 2004.
75. James Reilly, Kevin McCarthy, Lorraine McGinty, and Barry Smyth. Incremental critiquing. *Knowl.-Based Syst.*, 18(4-5):143–151, 2005.
76. James Reilly, Barry Smyth, Lorraine McGinty, and Kevin McCarthy. Critiquing with confidence. In *Proceedings of International Conference on Case-Based Reasoning (ICCBR)*, pages 436–450, 2005.
77. Paul Resnick and Hal R. Varian. Recommender systems. *Commun. ACM*, 40(3):56–58, 1997.
78. J. Rocchio. *Relevance Feedback in Information Retrieval*. G. Salton (editor), The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971.
79. U. Rohini and Vasudeva Varma. A novel approach for re-ranking of search results using collaborative filtering. In *Proceedings of the International Conference on Computing: Theory and Applications*, volume 00, pages 491–496, Los Alamitos, CA, USA, 2007. IEEE Computer Society.
80. Lior Rokach, Genetic algorithm-based feature set partitioning for classification problems, *Pattern Recognition*, 41(5):1676–1700, 2008.
81. Mehran Sahami and Timothy D. Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the International World-Wide Web Conference*, pages 377–386, 2006.
82. Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. In *Readings in information retrieval*, pages 355–364. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

83. J. Ben Schafer, Joseph Konstan, and John Riedi. Recommender systems in e-commerce. In *EC '99: Proceedings of the 1st ACM conference on Electronic commerce*, pages 158–166, New York, NY, USA, 1999. ACM Press.
84. U. Shardanand and P. Maes. Social Information Filtering: Algorithms for Automating "Word of Mouth". In *Proceedings of the Conference on Human Factors in Computing Systems (CHI '95)*, pages 210–217. ACM Press, 1995. New York, USA.
85. U. Shardanand and P. Maes. Social Information Filtering: Algorithms for Automating "Word of Mouth". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 210–217, 1995.
86. Xuehua Shen, Bin Tan, and ChengXiang Zhai. Implicit User Modeling for Personalized Search. In *Proceedings of the Fourteenth ACM Conference on Information and Knowledge Management (CIKM 05)*, 2005.
87. Alan F. Smeaton, Colum Foley, Daragh Byrne, and Gareth J. F. Jones. ibingo mobile collaborative search. In *CIVR*, pages 547–548, 2008.
88. Alan F. Smeaton, Hyowon Lee, Colum Foley, and Sinéad McGivney. Collaborative video searching on a tabletop. *Multimedia Syst.*, 12(4-5):375–391, 2007.
89. Barry Smyth. Case-based recommendation. In Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors, *The Adaptive Web*, pages 342–376. Springer-Verlag, 2007.
90. Barry Smyth. A community-based approach to personalizing web search. *IEEE Computer*, 40(8):42–50, 2007.
91. Barry Smyth, Evelyn Balfe, Oisín Boydell, Keith Bradley, Peter Briggs, Maurice Coyle, and Jill Freyne. A Live-user Evaluation of Collaborative Web Search. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI '05)*, pages 1419–1424. Morgan Kaufmann, 2005. Edinburgh, Scotland.
92. Barry Smyth, Evelyn Balfe, Jill Freyne, Peter Briggs, Maurice Coyle, and Oisín Boydell. Exploiting query repetition and regularity in an adaptive community-based web search engine. *User Model. User-Adapt. Interact.*, 14(5):383–423, 2004.
93. Barry Smyth, Peter Briggs, Maurice Coyle, and Michael P O'Mahony. A case-based perspective on social web search. In *Proceedings of International Conference on Case-Based Reasoning (ICCBR)*, 2009.
94. Barry Smyth, Peter Briggs, Maurice Coyle, and Michael P. O'Mahony. Google. shared! a case-study in social search. In *Proceedings of the International Conference on User Modeling, Adaptation and Personalization (UMAP)*, pages 494–508. Springer-Verlag, 2009.
95. Barry Smyth and Pierre-Antoine Champin. The experience web: A case-based reasoning perspective. In *Workshop on Grand Challenges for Reasoning from Experiences (IJCAI 2009)*, 2009.
96. Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. In *Document retrieval systems*, pages 132–142. Taylor Graham Publishing, London, UK, UK, 1988.
97. Micro Speretta and Susan Gauch. Personalized search based on user search histories. In *WI '05: Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 622–628, Washington, DC, USA, 2005. IEEE Computer Society.
98. Amanda Spink, Judy Bateman, and Major Bernard Jansen. Searching heterogeneous collections on the web: behaviour of excite users. *Information Research: An Electronic Journal*, 4(2), 1998.
99. Amanda Spink and Bernard J. Jansen. A study of web search trends. *Webology*, 1(2):4, 2004.
100. Amanda Spink, Dietmar Wolfram, Major B. J. Jansen, and Tefko Saracevic. Searching the Web: the Public and their Queries. *Journal of the American Society for Information Science*, 52(3):226–234, 2001.
101. Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 675–684, New York, NY, USA, 2004. ACM Press.
102. Jian-Tao Sun, Hua-Jun Zeng, Huan Liu, Yuchang Lu, and Zheng Chen. Cubesvd: a novel approach to personalized web search. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 382–390, New York, NY, USA, 2005. ACM Press.
103. Bin Tan, Xuehua Shen, and ChengXiang Zhai. Mining long-term search history to improve search accuracy. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 718–723, New York, NY, USA, 2006. ACM Press.

104. C. J. van Rijsbergen. *Information Retrieval*. Butterworth, 1979.
105. Xin Wang and Ata Kabán. State aggregation in higher order markov chains for finding online communities. In *IDEAL*, pages 1023–1030, 2006.
106. Xin Wang and Ata Kabán. A dynamic bibliometric model for identifying online communities. *Data Min. Knowl. Discov.*, 16(1):67–107, 2008.
107. Yiming Yang and Christopher G. Chute. An example-based mapping method for text categorization and retrieval. *ACM Trans. Inf. Syst.*, 12(3):252–277, 1994.
108. Baoyao Zhou, Siu Cheung Hui, and Alvis C. M. Fong. An effective approach for periodic web personalization. In *WI '06: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 284–292, Washington, DC, USA, 2006. IEEE Computer Society.



# 社会化标签推荐系统

Leandro Balby Marinho、Alexandros Nanopoulos、Lars Schmidt-Thieme、  
Robert Jäschke、Andreas Hotho、Gerd Stumme 和 Panagiotis Symeonidis

**摘要** 社会化标签系统作为新一代的网络应用已经成功建立了，并且在不断地发展。社会化标签系统本身具有开放性和社会性，这些特点已经证实能够有效地促进用户参与。然而在社会化标签系统带来更多机遇的同时，也使得一些旧的问题重新出现，如信息过载。推荐系统也是一种更好地提高相关内容级别同时过滤网上不断产生的“噪声”，方便获取有效信息的方式。但是社会化标签系统中，我们面临着新的挑战。用户感兴趣的不仅是内容，甚至包括标签和其他用户。加之，传统的推荐系统通常在二维数据上操作，而 STS(社会化标签系统，Social Tagging System)的数据要么通过三维数组来表示，要么由超边表示的超三角图(用户、资源、标签)来表示。在这一章里，我们系统总结了近年来在全新一代推荐系统上构建的社会化标签系统的最新学术成果。我们从以下三个方面着手总结：1)用于社会化标签系统的新的推荐系统方面，如用户、资源和标签推荐；2)解决 STS 数据三元特性的方法和算法；3)在现实的 STS 环境中部署推荐系统。我们同时对已有工作进行一个简单的对比，在这些对比的过程中指明新的研究方向。

## 19.1 简介

随着国内通信网络更加便宜快速、电子化设备更加低廉及互联网本身所具有的开放性，一种新的 Web 2.0 应用也雨后春笋般地出现了。这种网络的重要思想就是内容产生的方式更加分散，成本也更低，这促使整个网络向着更加开放互联民主的方向发展。在这一章里，我们主要关注 Web 2.0 里特殊的分支——社会化标签系统(简称 STS)。STS 赋予了每个普通人一个重要的角色，他们不仅发表和编辑内容，而且更重要的是，他们产生和分享轻量的元数据，这些元数据是用用户自由产生的关键字，这些关键字称为标签。由于用户通过标签和资源使得其在沟通和分享方面产生了最基本的条件，从而降低了协作的门槛，促进形成协同的轻量知识体系结构“大众分类法”<sup>①</sup>的产生。最著名的 STS 网站有 Delicious<sup>②</sup>、BibSonomy<sup>③</sup>和 Last.fm<sup>④</sup>等。Delicious 可以分享书的标签，BibSonomy 可以分享书的标

Leandro Balby Marinho, Alexandros Nanopoulos, Lars Schmidt-Thieme, Information Systems and Machine Learning Lab (ISMLL), University of Hildesheim, Marienburger Platz 22, 31141 Hildesheim, Germany, <http://www.ismll.uni-hildesheim.de>, e-mail: {marinho, nanopoulos, schmidt-thieme}@ismll.uni-hildesheim.de

Robert Jäschke, Andreas Hotho, Gerd Stumme, Knowledge & Data Engineering Group (KDE), University of Kassel, Wilhelmshöher Allee 73, 34121 Kassel, Germany, <http://www.kde.cs.uni-kassel.de>, e-mail: {jaeschke, hotho, stumme}@cs.uni-kassel.de

Panagiotis Symeonidis, Department of Informatics, Aristotle University, 54124 Thessaloniki, Greece, e-mail: symeon@csd.auth.gr

翻译：邓展成，佚名 审核：吴涛，胡聪(胡户主)

① 大众分类这个术语是大众和分类两个词的组合，也就是由普通用户产生并维护一个协同分类系统。

② <http://delicious.com/>

③ <http://www.bibsonomy.org/>

④ <http://www.last.fm/>

签和文献列表, Last.fm 则可以分享音乐。这些系统具有容易使用、用户免费自愿参与的特点。一旦一位用户登录了, 他就可以添加一个资源到这个系统, 同时可以为这个资源打上一个自定义标签。

如果一方面这一系列应用带来了新的机会, 另一方面则也会使得一些旧的问题重新复现, 也就是信息过载。上百万的用户和独立的提供者以一种不可控的方式产生了大量的内容和标签冲击着 STS, 这样降低了内容检索和信息分享的能力。最好的解决办法是在线上推荐系统中提高相关内容的级别用于减少不断产生的噪声影响。然而在 STS 中, 我们面临着一些新的挑战。用户关注的不只是发现内容, 还有标签甚至其他用户。加之, 传统推荐系统通常是在双向数组结构上完成的, 用大众分类法(folksonomy)数据要么通过三阶数组来表示, 要么由超边表示的超三角图(用户、资源、标签)来表示。而且大量文献中广泛研究基于显式用户反馈的评级预估, 例如, 用数值表示一个用户对某个物品的喜爱程度, 但在大众分类法里通常没有这些评估值。这也是我们开篇就开始讨论不用老方法去解决重新产生的问题的原因。在某种程度上我们需要分析传统的推荐系统是否适应 STS。

STS 是一个新的研究领域, 近来引起了很大的重视, 这通过发表的论文[15, 11, 37, 35, 31]数量不断在增加就能够体现出来, 当然 STS 也在不断地发展成长。实际上的大规模 STS, 如 Delicious、BibSonomy 和 Last.fm 也已经向他们的用户提供了相关的推荐服务, 这也体现出 STS 有着更广泛的商业用途。在这一章我们将以最简洁的方式总结最近在推荐系统上构建 STS 服务的最新工作: 1) 用于社会化标签系统的新的推荐系统方面, 如用户、资源和标签推荐; 2) 在现实的 STS 环境中部署推荐系统; 3) 解决 STS 大众分类法数据三元特性的方式和算法; 4) 标签获取的方式。同时重点对比已有的工作, 这样可以为我们的研究指明方向。

本章结构如下: 19.2 节描述大众分类法的特性, 同时分析传统推荐系统和 STS 的不同。19.3 节将讨论在实际 STS 应用中部署推荐系统的难题, 同时以 BibSonomy 为案例进行分析。19.4 节列出几类 STS, 如基于图/内容的用户、资源和标签推荐算法。19.5 节对比讨论第 18 章给出的算法。最后 19.6 节总结分析这个领域里新的研究方向。

## 19.2 社会化标签推荐系统

大众分类法是 STS 的基础构成, 由用户给内容标记标签进行内容分类。通常来说, 标签是通过分类来把内容分组, 按主题分类使其更加容易浏览。这是一个网站最基本的内容管理方式, 能够帮助用户发现他们感兴趣的内容。注意, 由于引进了标签这一概念, 以前传统推荐系统广泛采用的用户资源双向关联, 转变成由用户、资源和标签组成的三元关联。

由于标签是用户自愿主动提供的, 所以用户不愿意打标签和标签量迅速增加等问题也很容易暴露出来。在本章中我们将看到最可能解决这类问题的方法是建立 STS。如果对标签的利用合适, 那么标签也可以当作对内容个性化信息的一种补充, 最终能够提高资源型推荐系统的效果。在我们探究推荐系统如何利用标签提供的附加信息之前, 我们需要正式定义大众分类法和它的数据结构, 详尽阐述传统推荐系统与 STS 的区别, 以及在现实 STS 中部署推荐系统带来的挑战。这几个主题将在下面讲述。

### 19.2.1 大众分类法

**定义** 大众分类法可以用一个元组  $F=(U, T, R, Y)$  来表示, 其中

- $U$ 、 $T$  和  $R$  是非空有限集合，它们分别表示用户、标签和资源。
- $Y$  是  $U$ 、 $T$  和  $R$  的三元关系，其中  $Y \subseteq U \times T \times R$ ，称为标签分配。<sup>①</sup>

用户通常用 User ID 来表示，标签为一组随机字符串。资源的选择需要依赖业务系统。例如，Delicious 中的资源是 URL，而 BibSonomy 是 URL 和参考文献，Last.fm 中的资源是作曲家、音乐曲目和专集。

大众分类法有不同的表现方式。在 19.4 节将讲解每一种展示方式会产生不同的推荐算法。

**大众分类法表示成张量(Tensor)** 三元组集合  $Y$  可以由三维数组表示： $A = (a_{u,t,r}) < \mathbf{R} | U| \times |T| \times |R|$ 。张量  $Y$  有不同的表示方法(见图 19.1 左侧)。例如，Symeonidis 等[35]提出用稀疏矩阵来表示  $Y$ ，其中 1 表示正反馈，0 表示值为空：

$$a_{u,t,r} = \begin{cases} 1, & (u,t,r) \in Y \\ 0, & \text{其他} \end{cases}$$

另外，Rendle 等[26]为了学习个性化标签排序，区分了正/负样本和缺失值。这种思想是正负值产生于观察到的标签分配样本。观察标签分配是正反馈，而对已经打完标签的非观察标签分配为负反馈。其他情况被认为是空值。(见图 19.1 右侧)

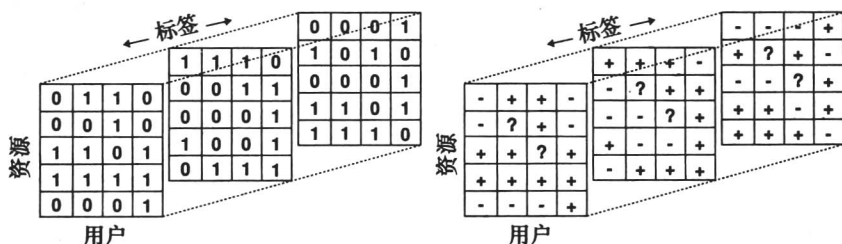


图 19.1 左图[35]: 0/1 稀疏矩阵表示表，其中 1 表示正反馈，0 表示其他数据。右图[36]: 已经标记资源的非观察标记分配是负例，其他为空

我们需要注意的是，不同于典型推荐系统，大众分类法没有一种数值评分来表示用户对某个给定的资源或标签的爱好程度。

**大众分类法表示成超图(hypergraph)** 相类似，超图也是大众分类法的直观表示，是一个三元无向超图： $G := (V, E)$ ，其中  $V := U \cup T \cup R$  是节点集合， $E := \{\{u, t, r\} | (u, t, r) \in Y\}$  是超图的集合(见图 19.2)。

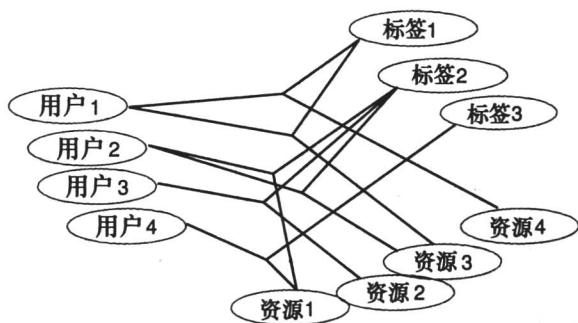


图 19.2 大众分类法的三元无向超图表示法

## 19.2.2 传统推荐系统范式

推荐系统是一个软件应用，其目标是基于用户信息来预测用户对某个特定资源的兴趣度，用户信息包括用户的历史访问购买信息、资源排名、点击流数据、人口统计信息等。

① 在最初的定义里[12]，额外引入了子标签/父标签的关系，我们在这里忽略了。这里采用的版本在形式概念分析[7]中称为三元环境[21, 34]。

推荐系统通常用来预测资源排名或者推荐给用户最喜欢的资源列表。传统来说, 给定  $m$  个用户和  $n$  个资源, 用户信息用一个稀疏用户资源矩阵  $\mathbf{X} \in \mathbf{R}^{m \times n} \cup \{.\}$  来表示, 其中  $\{.\}$  表示值为空。这个矩阵分解成多个行向量:

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]^T, \quad \mathbf{x}_u = [x_{u,1}, \dots, x_{u,n}], \quad u = 1, \dots, m$$

其中  $x_{u,r}$  表示用户  $u$  对资源  $r$  的评分,  $x_{u,r} \in \mathbf{R}$ 。每个行向量  $\mathbf{x}_u$  对应一个用户信息, 表示某个特定用户对资源的评分。这种矩阵分解经常采用一种利用用户与用户的相似度的算法, 例如, 著名的基于用户的协同过滤算法(CF)[27]。矩阵也可以通过列向量来表示:

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n], \quad \mathbf{x}_r = [x_{1,r}, \dots, x_{m,r}]^T, \quad r = 1, \dots, n$$

其中每一列向量  $\mathbf{x}_r$  对应的是  $m$  个用户对某一特定资源的评价。这种表示方式通常利用物品与物品之间的相似度, 采用相应的基于物品的协同过滤算法[3]。对于基于近邻的推荐算法的总结可详见第4章。

注意, 由于大众分类法的三方关系特性, 传统的推荐系统很难直接适应。因此为了开发出适合大众分类法的推荐系统, 需要满足以下条件之一: 1) 采用传统推荐算法把三元关系  $Y$  降维, 映射到低维空间(通常是二维矩阵)中; 2) 或者在三阶矩阵或三元无向超图上搭建一个新的推荐算法。如果采用方案一必须注意由于降维引起的重要数据的丢失, 这样会降低整个推荐系统的准确度。在 19.4 节将详细讨论这两类算法。

### 19.2.3 多模式推荐

不同于传统的推荐系统范式——它们关注的要么是排名预测, 要么是资源推荐, STS 用户可能对发现资源/标签甚至其他用户感兴趣, 还有这类推荐可以提供各种实体类型。

标签推荐已经用于 Delicious 和 BibSonomy 等系统中。其中包括基于其他用户在对相同资源提供的标签, 向用户推荐标签。标签推荐把物品的不同信息展示给用户, 同时也免除用户为了得到一些好的标签列表而需要做一些令人厌烦的工作。而且标签推荐可以减少用户不愿做标签引起的标签矩阵稀疏问题。图 19.5 说明 BibSonomy 系统中标签推荐。

有别于传统的推荐系统, 最需要注意的一点是, 它不是用于反复购买行为, 例如, 用户不会两次都买一本相同的书籍、电影、CD 等, 而重复性标签是 STS 的共同特点。标签可以用于标注一个资源, 也可以标记另一种不同的资源。也就是说, 传统的推荐系统只会向用户推荐用户未买过或未评分过的物品, STS 最终也会向用户推荐他在其他资源中使用过的标签。

资源推荐广泛用于电子商务和广告, 如亚马逊网站。随着 STS 越来越流行, 许多现行的资源推荐服务同样可以利用标签提高推荐的质量, 如基于其他类似用户的共同标签向用户推荐资源。在视频推荐网站 movielens<sup>①</sup> 上有一个著名的案例: 用户对自己喜爱的电影进行评分, 同时收到其他他们可能感兴趣视频的推荐。它起初是在典型的用户评分二维矩阵上完成相关传统推荐服务, 而最近添加了社会化标签特性, 在上面可以开发或部署对标签敏感的新算法[30]。

第三类推荐应用的重点关注向目标用户推荐感兴趣的资源, 这样可以帮助用户与有共同爱好的人建立关系, 鼓励他们贡献和分享更多的内容。在这里, 感兴趣的资源是指那些与目标用户有相似信息的资源。例如, 如果一个标签频繁被许多人使用, 那么这些人隐式

① <http://www.movielens.org>

地形成了一个有着共同兴趣的用户群，即使他们没有任何地理上或者网络上的联系。这个标签就可以标示这个用户群的共同兴趣。

每一种推荐模式，如标签、资源或者用户，都是有意义的，当然它们都需要依赖特定应用环境。有些算法能够提供集成多种模式推荐，它们与那些努力实现多种特定模式推荐系统是一样很吸引人的。

## 19.3 现实社会化标签推荐系统

### 19.3.1 有哪些挑战

对于一个推荐系统要在一个现实应用中取得成功，它一定要面临一些挑战。第一，所提供的推荐必须要符合应用场景。例如，标签应该描述被标注的资源和产品，应该能够吸引用户的兴趣，所荐资源也应该是令人感兴趣的并且是相关联的。第二，建议应该是可以描述的，这样用户更易知道为什么会向他推荐这些东西。第三，它们必须可以快速传递给用户，必须方便查看（例如，允许用户点击它们或者在输入标签时可以自动补全）。最后，推荐系统要保证推荐不要影响系统的正常使用。

在这一节里，我们把标签推荐作为 STS 推荐实例。大部分 STS 包含一个标签推荐器，在用户正在给某个资源作标注时，它把一些标签推荐给用户。正在被推荐的标签有着多种用途，例如，增加资源被标注的机会，提醒用户资源是什么，扩大用户的标签词典。而且，正如 Sood 等[33]指出标签推荐“从产生到认识都从根本上改变了打标签的过程”，这只需要更少认知努力和时间。

更正式地，给定一个用户  $u$  和一个资源  $r$ ，标签推荐任务是预测标签  $\text{tags}(u, r)$ ，该预测标签是该用户为一个资源分配的标签。我们用  $\hat{T}(u, r)$  表示推荐标签的（有序）集合。尽管在用户登录自己的账号时我们没有给标签排序，推荐系统给出的标签顺序对系统评估有着很重要的作用。

### 19.3.2 案例 BibSonomy

#### 19.3.2.1 系统简介

BibSonomy 起源于 Kassel 大学<sup>⊖</sup>知识与数据工程项目组 2005 年春季的一个学生项目。项目的目的是构建一个系统，该系统通过一种类似于 Delicious 书签方式把 BIBTEX[25]条目组织起来，Delicious 书签方式在那时就越来越受欢迎。BIBTEX 是一种流行的 LATEX [20]文献管理系统，因为许多研究人员用 LATEX 写科研论文。在把书签作为第二类资源整合到该系统之后直到项目完成，BibSonomy 在 2005 年底对外开放以供访问——初期只提供同事访问，后来在 2006 年正式全面开放。

图 19.3 是 BibSonomy 一个书签位的详细视图。第一行书签标题加粗，超链接是该书签的 URL。第二行是展示一条可选描述信息，用户可以把该描述信息分配给任何一篇文章。最后两行是一起的并且提供了详尽信息：其中包括用户赋给这篇文章的所有标签（web, service, tutori-

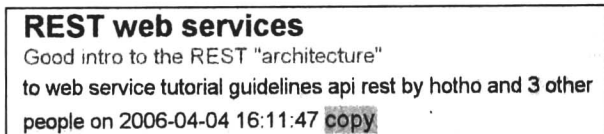


图 19.3 单个书签位的展示细节

⊖ <http://www.kde.cs.uni-kassel.de/>



al, guidelines 和 api); 以及给这篇文章打书签的某位用户的用户名(hotho)及多少用户给该文章作标签等信息。这部分都有超链接引导用户去往对应的标签页、用户页及展示该资源所有四篇文章的列表页——它列出 hotho 的文章及其他三位的文章。发布文章的页面结构也非常相似, 如图 19.4 所示。

19.3.2.2 BibSonomy 推荐

为了帮助用户更加方便地打标签, BibSonomy 含有标签推荐功能(图 19.5)。当用户发现了一篇有趣的页面(或者出版物), 张贴到 BibSonomy, 那么这系统会在发帖页上至少提供 10 条推荐标签。

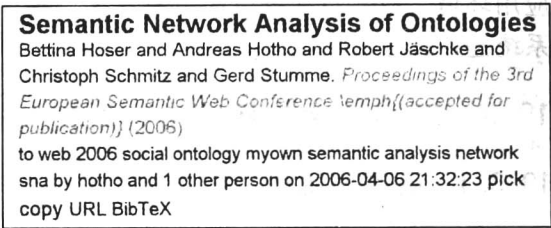


图 19.4 发布页的展示细节

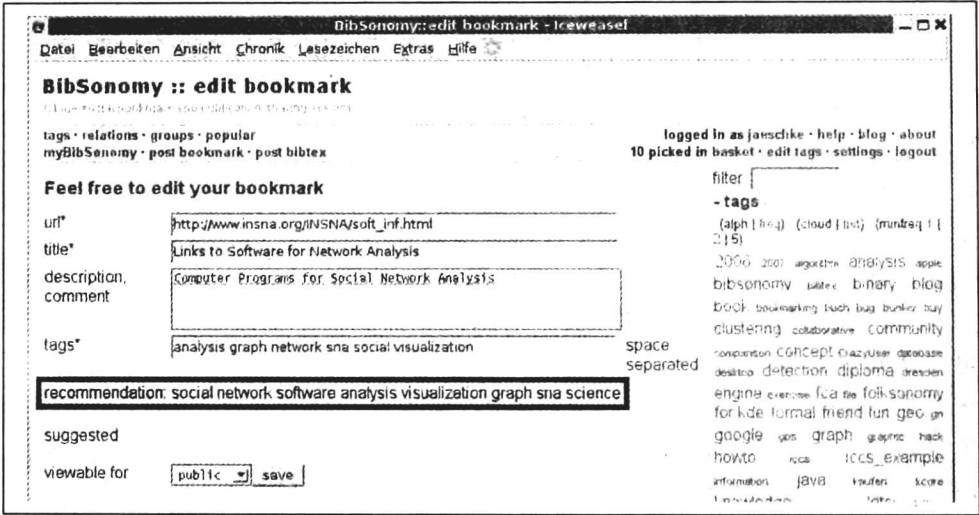


图 19.5 BibSonomy 书签标注过程的标签推荐

19.3.2.3 技术及基础设施的要求

为 BibSonomy 实现一个推荐服务需要处理如下几个问题:

首先, 推荐算法产生有效的推荐需要获得足够的数据是重要的必备条件之一, 它在设计阶段就需要得到解决。推荐服务需要访问用户正在发帖(例如, 资料填写完成并且正在通过 AJAX 技术加载推荐数据)的系统数据库。全部数据——如文档的全部文本——需要用于解决冷启动问题(如基于内容的推荐)。这种系统必须能够处理相当大的数据, 迅速选择出相关子集并且提供预处理。

已有的硬件条件和期望的数据量限制推荐算法的选择。尽管一些推荐算法允许推荐数据的(部分)提前计算。这就需要使用多余的内存, 在线上计算时可能得不到理想的结果。硬件和网络基础设施必须保证短时间内把推荐数据传递给用户, 而没有太大的延迟。同时需要提供一个简单无打扰的用户界面保证系统可用。

系统更高阶方面的需求也必须考虑, 例如, 用户事件的日志接入(如点击、键盘按键等), 用于有效地验证线上配置上已使用的推荐算法。同时也需要实时验证推荐效果, 使得能够手动调整结果选择策略去为用户或资源动态选择(目前)最好的推荐算法。多种算法连同简单包含外部推荐服务(通过一种开放推荐接口)的多路复用技术是 BibSonomy 系统里面最近开发的一种技术。

### 19.3.3 标签获取

标签的质量可以直接影响 STS 的推荐性能。尽管大众分类法代表的是群体智慧，社会标签带来了一系列问题，例如标签稀疏性(用户倾向于提供有限的标签数量)，一词多义(标签有多种解释结果)，或者标签特质(按用户自己的意思编写标签，如“未读”等)。所有这些问题都会影响推荐的质量。由于这个原因，我们换了一种方式获取标签。这能够帮助我们更好地分清社会标签流程中的优点和缺点。然后我们会检查下面这些标签获取方法的好坏。

- **专家标签：**这种方法主要是依赖一小部分该领域的专家，他们标注资源用途，主要是构造词汇。专家提供的标签客观，涵盖面广。Pandora<sup>⊖</sup>系统是一个著名的例子。专家在上面给音乐资源做标签。采用专家的好处是一组共同接受的标签词汇。当然，这需要大量的人工操作，费力耗时且成本高昂。
- **基于标注游戏的标签：**像 ESPGame<sup>⊖</sup>这样具有一定目的性的游戏(GWAP)[39]是一种突破性的想法，它是通过游戏使用户达到标注目的。两个参与者同时观察同张图片，要求他们填入标签，直到他们俩都填入了相同标签。伴随着 ESPGame 游戏的成功，其他类似应用在音乐领域的游戏(如 ListenGame<sup>⊖</sup>)也相继出现了。正如社会标签，游戏开发“人类的计算能力”。通过两人或者多人组合，这种标签结果集也有很高的准确度。这种游戏的问题是游戏参与者如果为了得高分，那么可能会降低标签质量。例如，他们也许在更具体的情况下填写更公共的标签，这样会增加匹配的概率。
- **基于内容的标签技术：**像 URL、音频等这样的资源含有丰富的内容。通过在网页上爬取相关信息并且把这些数据转换成适当的表示方式，标签可以通过数据挖掘算法产生。例如，在 ECML PKDD 发现频道 2008<sup>Ⓢ</sup>的标签推荐任务里，一些在测试集上预测的标签在测试集里面没有出现，这样需要强制用户采用资源的文本内容产生新的标签。在音乐领域，这种方法称为自动标签技术，通常被用于避免冷启动问题[5]。基于内容标签的好处是不需要用户直接参与这个打标签过程。坏处是这样的标签方式会产生噪声，计算密集型，用户被强制接受通过算法产生的标签。

对比上述方法，社会化标签技术有利于产生大规模标签数据集合。大量标签用户通常能够提高标签的质量。然而，社会化标签技术很容易出现冷启动问题，因为新的资源很少被打标签。在表 19.1 中我们总结了上述方法的优缺点。

表 19.1 标签收集方法的特点

方法	优点	缺点
社会化标签	大规模，群体智慧	个性化强烈，一词多义，冷启动
专家方式	标签准确	过程高昂，不适合大规模
游戏方法	群体智慧，可用于大规模	冷启动，易于操作
挖掘标签	自动化，避免冷启动	噪声，计算密集型

⊖ <http://www.pandora.com/>

⊖ <http://www.gwap.com/gwap/gamesPreview/>

⊖ <http://www.listengame.org/>

Ⓢ <http://www.ked.cs.uni-kassel.de/ws/rsdc08/>

尽管社会化标签技术很容易出现特质化、稀疏性和冷启动等问题,然而大量标签用户通常能够提高标签的质量。而且社会化标签系统(如同标注游戏一样)表示的用户计算范式有着巨大的能力去解决基于内容的推荐机制在标签获得上其本身无法解决的问题。但是不像计算机,人需要一些刺激才会成为这种通过标签推荐提供的集体计算的一部分。

## 19.4 社会化标签系统的推荐算法

正如我们在 19.2 节所指出的,有一些大众分类法的特性需要在设计推荐系统前加以考虑,如:

- 公众分类数据被表示成“张量”或者三重无向超图(tripartite undirected hypergraph)。因此,我们要么对数据做变换以适应传统的推荐算法,要么对现有算法进行扩展使其可作用于张量或者超图数据。
- 使用大众分类法的用户可能会对“多模式推荐”感兴趣,因此,那些支持所有模式、且在模式切换过程中不产生变化或者仅产生细微变化的算法是理想所需。
- 大众分类法支持多媒体资源。因而,基于内容的算法应能有效将内容信息整合入大众分类法的数据结构中。

在本节中,我们阐述社会化标签推荐系统中当下一些主流方法,展示和探讨它们是如何处理上述问题的。

### 19.4.1 协同过滤

协同过滤是最常用并成功地应用于个性化系统的方法之一,有着大量且持续活跃文献(见第 4 章和第 5 章)。大体上说,这种算法匹配出兴趣相同的人来进行推荐。正如 19.2.2 节中指出的,传统的推荐系统通常运用于“表示用户与资源之间二元关系”的二阶张量上。因此,由于大众分类法的三元关系性质,传统的协同过滤算法不能直接应用,除非三元关系被降到一个低维空间[24]。为此,就“基于用户的协同过滤”来说,矩阵可以选择以下两个 2 维映射之一: 1) 若存在  $t \in T$ , 使得  $(u, t, r) \in Y$ ,  $\pi_{UR} Y \in \{0, 1\}^{|U| \times |R|}$ , 则  $(\pi_{UR} Y)_{u,r} := 1$ , 否则为 0; 2) 若存在  $r \in R$ , 使得  $(u, t, r) \in Y$ ,  $\pi_{UT} Y \in \{0, 1\}^{|U| \times |T|}$ , 则  $(\pi_{UT} Y)_{u,t} := 1$ , 否则为 0。参见图 19.6。最后,你也可以考虑“资源-标签”投影矩阵,这将引向“非个性化的基于内容的模型”。

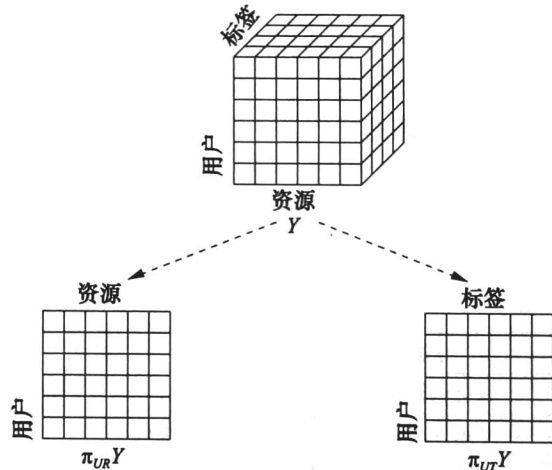


图 19.6 映射到“用户的资源”和“用户的标签”空间

这些映射保留了用户信息,引向基于“资源-用户”或者“标签-用户”的推荐系统。

请注意,在这里,形成“一个用户的  $k$  近邻” $N_k^u$  有两个可能的设定方法,以资源作为对象或者以标签作为对象。有了定义好的矩阵  $X$ ,并决定了使用  $\pi_{UR} Y$  或  $\pi_{UT} Y$  来用于计算用户的近邻,我们就有了应用协同过滤方法所需的设置。首先,我们基于行分解的矩阵  $X$

和一个给定  $k$ ，来计算和用户  $u$  最相似的  $k$  个用户的集合  $N_u^k$ ：

$$N_u^k := \underset{v \in U \setminus \{u\}}{\operatorname{argmax}}^k \operatorname{sim}(\mathbf{x}_u, \mathbf{x}_v) \quad (19.1)$$

其中， $\operatorname{argmax}$  函数的上标表示要返回的近邻数  $k \in \mathbb{N}$ ； $\operatorname{sim}$  是任何含义清晰的相似性度量，

例如，通常的余弦相似性度量  $\operatorname{sim}(\mathbf{x}_u, \mathbf{x}_v) := \frac{\langle \mathbf{x}_u, \mathbf{x}_v \rangle}{\|\mathbf{x}_u\| \|\mathbf{x}_v\|}$ 。

**多模式推荐**(Multi-mode Recommendation) 有了计算好的近邻，我们就可以为一个给定的用户  $u$ ，一个给定的资源  $r$  和某个  $n \in \mathbb{N}$  提取出有  $n$  个推荐标签的集合  $\hat{T}(u, r)$ ，如下：

$$\hat{T}(u, r) := \underset{t \in T}{\operatorname{argmax}}^n \sum_{v \in N_u^k} \operatorname{sim}(\mathbf{x}_u, \mathbf{x}_v) \delta(v, t, r) \quad (19.2)$$

其中，若  $(v, t, r) \in Y$ ，则  $\delta(v, t, r) := 1$ ，否则为 0。

如果你想推荐资源，可以使用类似的方法。请注意，如果我们只使用  $\pi_{UR} Y$  投影，运用标准的基于用户的协同过滤算法就可以了(见式(19.3))。但既然标签能够提供有关用户兴趣的额外信息，它们最终可以提升推荐的质量，就应当加以利用。一个很简单的感知标签的推荐方法是基于  $\pi_{UT} Y$  投影矩阵来计算用户近邻，并聚合这些近邻的资源来生成推荐列表。文献[6]提出了一个类似想法，通过提取资源的推荐列表，第一次采用“用户-标签”投影矩阵  $\pi_{UT} Y$  来计算出一个标签的有序列表。然而，仅单独使用  $\pi_{UT} Y$ ，所述资源信息被丢弃了；在这里，资源信息是用户兴趣的关键模式。在这个意义上说，我们需要找到一种方法，以适应 folksonomy2 路数据结构中的所有三种模式，从而可以应用标准的协同过滤。Tso-Sutter 等[37]提出了一种方法，利用标签将典型的“用户-资源”矩阵扩展出伪用户和伪资源(见图 19.7)。请注意，在这种方式中，用户/资源信息(user/resource profile)被自动用标签加以充实；该文接着提出一种融合算法，在扩展矩阵上合并基于用户的协同过滤(UCF)和基于物品的协同过滤(ICF)的预测。回想一下，在用标准的基于用户的协同过滤进行资源预测时，一个给定用户  $u$  对某个资源  $r$  的兴趣度分值计算为：与资源  $r$  共现的近邻用户的归一化数目，

$$p^{\text{ucf}}(x_{u,r} = 1) := \frac{|\{v \in N_u \mid x_{v,r} = 1\}|}{|N_u|} \quad (19.3)$$

对于用基于物品的协同过滤来进行资源预测，文献[3]提出了算法：一个给定用户  $u$  对某个资源  $r$  的兴趣度分值计算为： $r$  和“与  $u$  共现的  $r$  那些近邻资源”的相似度的平均总和

$$p^{\text{icf}}(x_{u,r} = 1) := \sum_{(r' \in N_r \mid x_{u,r'} = 1)}^n \operatorname{sim}(r, r') \quad (19.4)$$

融合以上两种分值得到

$$p^{\text{iucf}}(x_{u,r} = 1) := \lambda \cdot \frac{p^{\text{ucf}}(x_{u,r} = 1)}{\sum_r p^{\text{ucf}}(x_{u,r} = 1)} + (1 - \lambda) \cdot \frac{p^{\text{icf}}(x_{u,r} = 1)}{\sum_r p^{\text{icf}}(x_{u,r} = 1)} \quad (19.5)$$

其中， $\lambda$  只是一个控制 ucf 和 icf 影响比例的参数。请注意，由于 ucf 和 icf 计算出的预测列表的分值以不同的单位来度量(ucf 用的是物品的频度，而 icf 用的是物品的相似度)，预测分值做了归一化。对于某个  $n \in \mathbb{N}$ ，Top- $N$  推荐列表接着这样生成，

$$\arg \max_r^n p^{\text{iucf}}(x_{u,r} = 1) \quad (19.6)$$

Wetzker 等[41]提出了类似的想法，对概率潜在语义分析(PLSA)模型[10]进行标签

扩展来做资源推荐。在标准 PLSA 中, 一个资源和一个给定用户共现的概率可算作

$$P(r|u) := \sum_z P(r|z)P(z|u) \quad (19.7)$$

其中,  $Z := \{z_1, \dots, z_q\}$  是一个隐藏主题变量, 它被假定为观察到的用户和资源之间共现分布的起源。于是假定资源与标签共现的起源是同样的隐藏主题。

$$P(r|t) := \sum_z P(r|z)P(z|t) \quad (19.8)$$

两个模型有公共因子  $P(r|z)$ , 通过最大化对数似然函数

$$L := \sum_r \left[ \lambda \sum_u f(r, u) \log P(r|u) + (1 - \lambda) \sum_t f(r, t) \log P(r|t) \right] \quad (19.9)$$

将两个模型结合起来。其中,  $f(u, r)$  对应资源和用户的共现次数,  $f(r, t)$  对应资源和标签的共现次数,  $\lambda$  是一个预先定义的权重, 用于平衡各个模型的影响。常见的期望最大化 (Expectation-Maximization, EM) 算法用来对模型做最大似然估计。对于一个给定的用户  $u$ , 各个资源按照概率  $P(r|u)$  (见式 (19.7)) 赋权并排序, 排序靠前的资源最终被推荐。

如果要推荐用户, 我们可以基于  $\pi_{UT} Y$  或  $\pi_{UR} Y$  来推荐近邻用户。考虑 folksonomy 的三种模式来推荐近邻, 你可以, 举例说, 使用 [37] 提出的矩阵扩展 (参见图 19.7), 或者基于“用户-资源”和“用户-标签”映射矩阵计算用户相似度的线性组合。

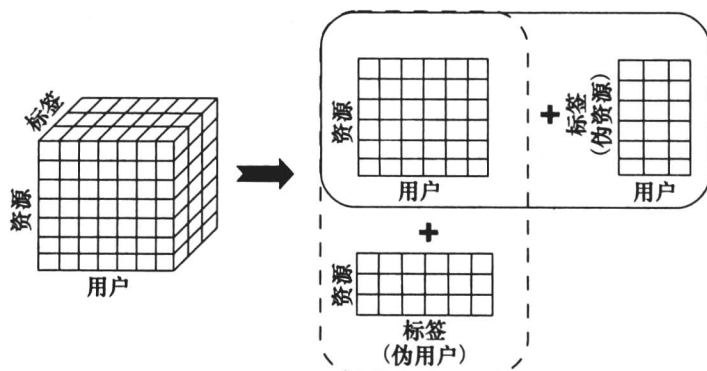


图 19.7 扩展“用户-资源”矩阵：水平方向加入标签作为伪资源和垂直方向加入标签作为伪用户

**关于复杂性** 鉴于整个输入矩阵需要保存在内存中, CF 通常面临扩展性问题。在 STS, 你可能不得不最终在内存中保留一个以上矩阵, 保留几个取决于你想要使用哪些映射。为了做推荐计算, 我们通常需要进行三个步骤:

1. 映射计算: 为了得到映射, 我们需要确定使用  $(u, r)$ 、 $(u, t)$  或  $(r, t)$  共现。对于这一点, 我们需要在  $Y$  上做线性扫描。

2. 近邻计算: 在传统的基于用户的 CF 算法中, 计算近邻  $N_u$  的复杂度和用户数量呈线性关系, 因为你需要计算一个给定的测试用户和数据库中所有其他用户的相似度。此外, 我们还要按相似程度排序, 以确定  $K$  近邻。

3. 进行推荐: 对一个给定测试用户做 Top- $N$  标签(或资源)推荐, 我们需要: 1) 计算标签(或资源)与近邻用户  $N_u$  的共现次数; 2) 依据与相应近邻用户的相似程度, 对每个共现赋权; 3) 根据标签(或资源)的权重对其排序(例如式 (19.2))。

#### 19.4.2 基于排序的推荐

下面介绍基于排序分值的推荐算法, 这点是从网页排序受到启发的。它们共同的特点



是,分值是依据从底层的大众分类数据结构中提取光谱属性(spectral attributes)来计算的。然而,用不同的方式表示大众分类(见 19.2.1 节)会导致不同的基于排序的算法。

#### 19.4.2.1 基于张量分解进行排序

以张量表示  $Y$ , 就可以利用由用户、标签和资源形成的多路关联(multi-way correlations)的  $A$  内底层潜语义结构来做推荐。这可以通过基于张量分解的推荐算法来实现, 见文献[26, 35, 43]。这种算法可以有效地检测出多路关联, 带来性能改善。第 5 章阐述了几个用于评分预测问题的最先进的矩阵分解(即二阶张量分解)方法。

$A$  的分解表示如下(见图 19.8):

$$\hat{A} = \hat{C} \times_u \hat{U} \times_r \hat{T} \times_l \hat{R} \quad (19.10)$$

其中  $\hat{U} \in \mathbf{R}^{|U| \times k_U}$ ,  $\hat{T} \in \mathbf{R}^{|T| \times k_T}$ ,  $\hat{R} \in \mathbf{R}^{|R| \times k_R}$  分别是表示用户、资源、标签的低秩特征矩阵, 其中潜在维度  $k_U$ 、 $k_R$ 、 $k_T$  较小;  $\hat{C} \in \mathbf{R}^{k_U \times k_R \times k_T}$  是一个表示潜在因素之间的相互作用的张量, 称为核心张量。符号  $\times_i$  表示张量和矩阵的  $i$  模乘积。于是待学习的模型参数是四元组  $\theta = (\hat{C}, \hat{U}, \hat{R}, \hat{T})$ 。这种分解是指被称为 Tucker 分解(Tucker Decomposition, 塔克分解)[18]的通用分解模型。参数学习之后, 可以如下进行预测:

$$\hat{a}_{u,t,r} = \sum_{\tilde{u}} \sum_{\tilde{r}} \sum_{\tilde{t}} \hat{c}_{\tilde{u}, \tilde{r}, \tilde{t}} \cdot \hat{u}_{u, \tilde{u}} \cdot \hat{r}_{r, \tilde{r}} \cdot \hat{t}_{t, \tilde{t}} \quad (19.11)$$

其中, 特征矩阵的特征维度坐标(indices over the feature dimension)标了一条波浪线, 特征矩阵的元素标了一顶帽子(如  $\hat{t}_{t, \tilde{t}}$ )。

Symeonidis 等[35]提出使用 0/1 的解释方案(见图 19.1 左部)通过高阶 SVD(HOSVD)来分解  $A$ ; 高阶 SVD 是张量 SVD 的多维模拟, 近期综述参见文献[18]。基本的想法是通过优化平方损失来最小化  $\hat{A}$  各个元素上的损失, 即

$$\operatorname{argmin}_{\hat{\theta}} \sum_{(u,t,r) \in U \times T \times R} (\hat{a}_{u,t,r} - a_{u,t,r})^2$$

Rendle 等[26]从另一个角度提出 RTF(利用张量分解进行排序), 这种方法学习出  $A$  的最优分解来解决标签推荐这个特定问题。首先, 将观察到的标签赋值分为积极的、消极的和缺失的, 具体描述参见 19.2.1 节(见图 19.1 右部)。令  $P_A = \{(u, r) \mid \exists t \in T: (u, t, r) \in Y\}$  是  $Y$  中所有不同的“用户/资源”组合的集合, 一个特定的  $(U, R) \in P_A$  的积极标签和消极标签的集合定义为

$$T_{u,r}^+ = \{t \mid (u, r) \in P_A \wedge (u, t, r) \in Y\}$$

$$T_{u,r}^- = \{t \mid (u, r) \in P_A \wedge (u, t, r) \notin Y\}$$

由此, 标签成对排序(pairwise tag ranking)的约束条件可以定义为

$$a_{u,t_1,r} > a_{u,t_2,r} \Leftrightarrow (u, t_1, r) \in T_{u,r}^+ \wedge (u, t_2, r) \in T_{u,r}^- \quad (19.12)$$

于是, 代替基于 HOSVD 的方法中最小化最小二乘误差, 提出了另一个优化准则: 最大化排序统计量 AUC(ROC 曲线下面积)。特定  $(u, r) \in P_A$  的 AUC 度量, 定义为

$$\text{AUC}(\hat{\theta}, u, r) = \frac{1}{|T_{u,r}^+| |T_{u,r}^-|} \sum_{t^+ \in T_{u,r}^+} \sum_{t^- \in T_{u,r}^-} H_{0.5}(\hat{a}_{u,t^+,r} - \hat{a}_{u,t^-,r}) \quad (19.13)$$

其中,  $H_{0.5}$  是 Heaviside 函数:

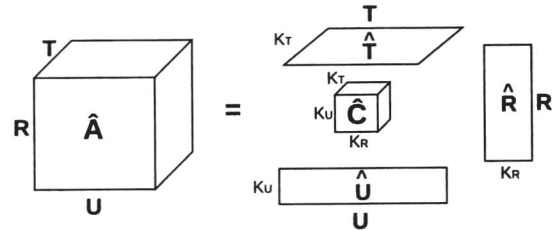


图 19.8 张量分解

$$H_a := \begin{cases} 0, & x < 0 \\ \alpha, & x = 0 \\ 1, & x > 0 \end{cases} \quad (19.14)$$

关于排序统计量 AUC 和所观察到的数据的整体优化任务可以表示为

$$\operatorname{argmax}_{\hat{\theta}} \sum_{(u,r) \in P_A} \text{AUC}(\hat{\theta}, u, r) \quad (19.15)$$

该优化问题于是可以通过梯度下降法[26]来求解。注意, 既然最大化仅通过所观察到的标签赋值来实现, 用这个优化标准时缺失值也被考虑在内。

**多模式推荐**(Multi-mode Recommendation) 一旦  $\hat{A}$  计算出来了, 一个给定的用户  $u$  和一个给定的资源  $r$  的  $n$  个最高分标签的列表可以这样计算:

$$\text{Top}(u, r, n) := \operatorname{argmax}_{t \in T}^n \hat{a}_{u,t,r} \quad (19.16)$$

给定标签  $t$  和用户  $u$  去推荐  $n$  个资源, 可以用以上类似的方式(见文献[36])。这样看, 张量建模能够以简单的方式支持进行多模式推荐。然而, 以上所述的 RTF 方法中, 矩阵分解被用来解决一个特定的标签排序优化问题, 可能有必要为每个兴趣模式定义特定的优化函数。

**关于复杂度** 像 RTF 或 HOSVD 这类分解模型的一个主要的好处是, 模型构建后, 预测只依赖于较小的分解维度。HOSVD 可以用 Sun 和 Kolda[19]的方法有效执行。还有一些方法通过切片[38]或近似[4]来改善大型数据集的可扩展性。

#### 19.4.2.2 FolkRank

网络搜索算法 PageRank[2]反映出这样的思想: 如果一个网页有很多网页可以链接到它, 且那些页面本身是重要的, 那么这个网页是重要的<sup>①</sup>。在文献[12]中, hotho 等采用相同的基本原则进行类似 Google 的大众分类的搜索和排序。FolkRank 算法的核心思想是, 一个被重要用户打上重要标签的资源本身就变得重要起来。同理, 这对标签和用户也适用。于是, 我们构建一个图, 图的顶点在互相传播它们的权重中加强彼此。在本节中, 我们简要地回顾 FolkRank 算法的原理, 并解释它如何用于生成标签推荐。

由于大众分类与 Web 图的性质不同(无向三元超边, 而不是定向二元边), PageRank 不能直接应用于大众分类。为了在大众分类上运用权重传播方案, 我们分两步来解决这个问题。首先, 我们将超图转换成一个无向图。然后, 我们采用一种差分排序的方法, 来应对网络的扭曲结构和大众分类的无向性, 并支持特定主题排序(topic-specific rankings)。

**PageRank-适应大众分类** 首先, 我们将大众分类  $F=(U, T, R, Y)$  转化成一个无向三分图  $G_F=(V, E)$ 。图的节点集  $V$  由标签、用户和资源的不相交集合组成(如,  $V=U \cup T \cup R$ )。所有标签和用户、用户和资源、标签和资源的共现都成为各自节点之间的边; 即  $Y$  中的每个三元组  $(U, T, R)$  产生出  $E$  中的三条无向边  $\{U, T\}$ 、 $\{U, R\}$ 、 $\{T, R\}$ 。

类似于 PageRank, 我们采用随机冲浪模型; 这个模型基于这样的想法: 一个理想化的随机网页访问路径沿着链接进行游走(比如, 从资源页面标签跳转到标签页面或者用户页面), 但偶尔会不沿着链接(随机)跳跃到一个新的网页上。这引出下面的定义。

图中顶点的排序(像在 PageRank 中那样)通过“权重扩散”来计算:

$$w_{t+1} \leftarrow dA^T w_t + (1-d)p \quad (19.17)$$

其中,  $w$  是权重向量,  $V$  中的每个节点对应向量中一个元素;  $A$  是上边定义的图  $G_F$

<sup>①</sup> 这个想法以相似的方式在 HITS[17]中拓展到网络双向子图和[42]中的  $n$  元有向图。

的邻接矩阵<sup>⊖</sup>的行随机版本； $p$  是一个随机冲浪矢量，在我们的设定中用作偏好向量； $d \in [0, 1]$  用于决定  $p$  的影响强度。归一化向量  $p$  来强制满足等式  $\|w\|_1 = \|p\|_1$ 。这<sup>⊖</sup>确保了系统中的权重将保持不变。每个节点的排序分值是其在迭代过程中的极限  $w_i = \lim_{t \rightarrow \infty} w_i$ 。

对于全局排序，有人会选择  $p = \mathbf{1}$ ，即由 1 组成的向量。然而，为了生成推荐，可以调整  $p$ ，对当前要生成推荐的那个用户节点和资源节点给予更高的权重。然后，推荐结果  $\hat{\Gamma}(u, r)$  就是排在前  $n$  个“标签节点”组成的集合。

由于图  $G_F$  是无向的， $t$  时刻经过某条边的权重大部分会在  $t+1$  时刻回流。于是，计算结果与完全基于度数的排序颇为相似（但由于随机冲浪，并不完全相同）。在后面的实验中，我们将看到这个算法版本表现得合理，但不出色。这符合文献[12]中的观察，该文表明特定主题的排序(topic-specific rankings)会因全局图的结构而产生偏差。因此，我们在文献[12]研发了下面这个差分方法。

**FolkRank-特定主题排序** 不管使用什么偏好向量，图  $G_F$  的无向性使得它很难使高边度节点以外的其他节点获得高排名。FolkRank 的差分方法解决了这个问题，它给 folksonomy 中的元素计算一个特定主题的排序。就我们的情况而言，主题是由我们打算做标签推荐的用户/资源对  $(u, r)$  决定的。

1. 令  $w^{(0)}$  为式(19.17)在  $p = \mathbf{1}$  时的不动点。
2. 令  $w^{(1)}$  为式(19.17)在  $p = \mathbf{1}$  但  $p[u] = 1 + |U|$  且  $p[r] = 1 + |R|$  时的不动点。
3.  $w := w^{(1)} - w^{(0)}$  即最终的权重向量。

于是，与没有偏好向量的基准方法相比，我们计算出给定一个用户/资源对时，节点间的相互加强后的赢家和输家。我们称大众分类的元素  $x$  的最终权重  $w[x]$  为  $x$  的 FolkRank。<sup>⊗</sup>

**多模式推荐(Multi-mode Recommendation)** 对一个给定的用户/资源对  $(u, r)$  生成标签推荐，如前所述计算排序，然后限制结果集  $\hat{\Gamma}(u, r)$  到前  $n$  个标签节点。同样，你可以通过给某个用户(或资源)优先权，给用户(或资源)计算推荐结果。由于 FolkRank 在大众分类所有三个维度上计算排序，这将产生对给定用户(或资源)最相关的标签、用户和资源。

**关于复杂度** 改进的 PageRank 的一轮迭代需要  $dAw + (d-1)p$  次计算，其中， $A \in \mathbb{R}^{s \times s}$ ， $s := |U| + |T| + |R|$ 。假设  $t$  表示迭代次数，复杂度为  $(s^2 + s)t \in \mathcal{O}(s^2 t)$ 。然而，由于  $A$  是稀疏的，更有效的做法是线性地遍历  $Y$  中的所有标签赋值来计算乘积  $Aw$ 。计算排序后，我们需要将标签的权重排序，以收集前  $n$  个标签。

### 19.4.3 基于内容的社会化标签推荐系统

目前为止，讲到的所有算法都不利用资源的内容，因此可以应用到任何大众分类，无须考虑支持的资源的类型。尽管如此，资源的内容是一种有价值的信息来源；特别是在冷启动的情况下，用户显式反馈稀缺。下面简短地讨论显式使用资源内容的推荐系统。

⊖ 如果  $\{i, j\} \in E$ ，则  $a_{ji} := \frac{1}{\text{degree}(i)}$ ，否则为 0。

⊗ 也包括没有排序降级的情况，在无向图  $G_F$  中都保持为零。

⊗ 文献[12]表明， $w$  在真实世界的一个大规模数据集上产生了真正有价值的结果；而  $w^{(1)}$  产生了与主题相关的元素和高边度元素的非结构化混合。在文献[13]中，我们采用这种方法用于检测大众分类随时间推移的趋势。

### 19.4.3.1 基于文本

Song 等[32]提出了一种基于图聚类的方法来给文本资源打标签,如网页或其他类型的文档。它不做个性化推荐,因为它不单独考察各个用户。特别是,它考虑资源中包含的文档、标签、词语之间的关系。这些关系用两个二分图表示。该方法分作两个阶段:

- 在离线阶段,它有效地对两个二分图的加权邻接矩阵进行低秩逼近,采用 Lanczos 算法[8]将图对称分割为多类簇。此外,它还提出了一种新的节点排序方案,对簇内标签对应的节点进行排序。其次,它采用一个泊松混合模型来给每个类别学习文档分布。
- 在在线阶段,给定一个文档向量,基于标签和文档的联合概率,根据它们(标签)的簇内排序给这个文档推荐标签。

如文献[32]中所解释的,这个双阶段框架可以被解释为一种“无监督—有监督”学习过程。在离线阶段,节点被分割成簇(无监督学习)和簇标签被分配给文档节点,作为“类”的标签。此外,标签节点在各个簇内被赋上排序。然后,基于文件和词语节点的分布建立起了一个混合模型。在线阶段,一个文档被分类(有监督学习)到预定义的簇上(这些簇是在第一阶段利用朴素贝叶斯得到的),使得标签按它们的排序从高到低进行推荐。

Song 等[32]强调方法的效率,泊松混合模型可以保证这一点,它让推荐在线性时间内完成。从 CiteULike(9623 篇论文和 6527 个标签)和 Delicious(22 656 的 URL 和 28 457 标签)爬取的两个大数据集上的实验结果显示,推荐可以在一秒钟内给出。

最近 Illig 等[14]也研究了“给定一个资源,基于内容来推荐标签”的不同方法。

### 19.4.3.2 基于图像

Abbasi 等[1]提出使用标签作为高层次特征,同低层次的图像特征一起在 Flickr 上训练图像分类器。尽管这种方法并不直接适用于推荐系统,它提供了一些关于“如何结合标签和低层次图像特征”的有趣见解,这可以最终作为一个推荐的输入。想法是,先从图像的标签信息创建一个向量空间,然后使用小波变换创建一个低层次的图像特征空间。接着合并这两个特征空间,并用来在组合的特征空间中训练一个单类支持向量机(SVM)分类器。

为了创建一个基于标签的图像特征向量,用“词袋”(bag-of-words)模型来将标签表示成特征。然后,对特征向量按词频(Term Frequency, TF)做归一化。注意,如果所属图像有很多标签,那么标签的权重会较低。接着,标签特征向量可以表示为

$$f_t := (tf(t_1, r), tf(t_2, r), \dots, tf(t_{|T|}, r))^T$$

其中,  $tf(t, r)$  表示与资源  $r$  共现的标签  $t$  的归一化词频。

RGB 颜色用于低层次特征提取。每个图像  $r$  被表示为一个四维向量

$$f_r := (c_{1,r}, c_{2,r}, c_{3,r}, c_{4,r})^T$$

其中,第一个成分  $c_{1,r}$  是图像  $r$  的像素值均值,其余成分分别代表红色、绿色和蓝色通道的像素值均值。

然后,特征向量结合起来,即

$$f_{t,r} := (tf(t_1, r), tf(t_2, r), \dots, tf(t_{|T|}, r), c_{1,r}, c_{2,r}, c_{3,r}, c_{4,r})^T$$

这种组合的特征向量接着用来作为单类 SVM 分类器的输入。从 Flickr 收集的真实数据上的实验表明,使用组合特征向量训练的分类器的性能大大优于只用标签特征向量或低

层次图像特征向量单独训练的分类器。

#### 19.4.3.3 基于音频

Eck 等[5]提出了一种直接从 MP3 文件预测社会化标签的方法。这些标签被称为自动标签(automatic tags, 简称 autotags), 因为它们直接产生于音乐内容。当一个集合中存在未贴标签或者标签低劣的几首歌曲, autotags 就能起到作用。因此, autotags 在音乐推荐系统中帮助解决“冷启动问题”。尽管如此, autotags 通过给音乐推荐系统的所有曲目提供一组具有可比性的基准标签来平滑标签空间。

Autotags 通过一个机器学习模型生成, 使用元学习算法 AdaBoost 从音频特征中预测标签, 例如, 梅尔频率倒谱系数(MFCC)特征, 为歌曲中选定标签计算的自相关系数, 通过常量 Q 频率(constant-Q frequency)采样的频谱系数。音频特征在歌曲内的短滑动窗口上计算, 产生了数量巨大的特征。为了减少这个数量, 通过在短窗口(如 5 秒)特征数据上计算之前音频特征单独的均值和标准差(即独立高斯), 创建出“聚合”特征。

特征选择按如下方式进行。模型基于“各个特征最小化经验误差的能力”来选择特征。因此, 与之关联的弱学习器(learner)被 AdaBoost 选中太晚, 这样的特征就被丢弃掉。

由于标签频率的高度偏斜分布(high skewness), 预测任务被视为一个分类问题。对于每一个标签, 预测一个特定的歌手“无”(none)、“有一些”(some)或“有很多”(a lot)某个特定的标签(相比其他标签)。3 个类别(即“无”“有一些”“有很多”)之间的边界的量化过程: 将所有歌手归一化的标签数目求和, 为每个标签产生一个 100 组距的直方图, 移动类别边界使得相等数目的歌手分到每个类别中。

要生成自动标签, 这些类别需要转化成“词袋”来与歌手关联上。基于 3 个类别的语义, 从“a lot”组距中减去“none”组距的值, 因为“none”与“a lot”是相反的(因此, “some”类只是为了使分类器在预测“none”和“a lot”时更具有选择性)。

在 1277 位艺术家的 89 924 首歌曲上进行了自动标签生成的实验评估, 产生出超过 100 万个 5 秒聚合特征。重点关注在线社交电台 Last.fm 上的 60 个最热门标签。这些标签包括如“Rock”这样的流派和像“chillout”这种与情绪有关的标签。分类错误显著低于随机误差。正如文献[5]所述, 性能应该与其他分类器(如 SVM 和神经网络)做对比来更好地评估该方法的优点。

#### 19.4.4 评估方案和评估度量

下面给出了一些用于评价不同的推荐模式的评估方案与度量。想要更全面地了解推荐系统的评估方案, 请参阅第 8 章。

**资源推荐** 为了评估标签感知(tag-aware)的资源推荐, 用于传统的推荐系统的常见方案和度量, 可以直接使用[9](参见第 8 章)。

**标签推荐** 对于评估标签推荐, 有两种可能的场景, 这最终会导致两种不同的评估方案:

- “新发布”: 用户选择了一个他尚未标记的资源, 系统尝试为这个资源向用户推荐一个包含  $n$  个标签的个性化列表。该方案最初在[24, 16]中使用, 称为弃一(LeaveOnePostOut)方案。对每一个用户  $u$ , 随机选择他的一个资源, 删除该资源的所有标签, 从而生成一个单独的训练集。于是, 训练集是  $\text{folksonomy}(U, T, R, Y')$ ,  $Y' := Y \setminus (\{u\} \times \text{tags}(u, ru) \times \{ru\})$ , 其中,  $\text{tags}(u, ru)$  是一组如用户  $u$  给  $ru$  打的标签。该任务随后就产生与  $\text{tags}(u, ru)$  相近的预测。这就再现了



这样的场景：一个用户有一个未标记的资源，系统尝试给出有用的推荐。

- “标签提炼”：用户选择一个他已经标记过的资源，系统提出了  $n$  个额外标签的个性化列表，用户可能想要用来改善他对资源的标记。[35]首次引入这个方案。想法是将标签赋值拆分成过去的标签赋值(训练集)和未来的标签赋值(测试集)。这反映了这样的场景：用户在逐步给物品打标签，在他们提供所有标签之前接收到推荐。

对于这两个方案，常见的准确率、召回率和  $f1$ -测度经常被使用[15, 35, 11, 26]：

$$\text{Recall}(\hat{T}(u, r_u)) = \frac{|\text{tags}(u, r_u) \cap \hat{T}(u, r_u)|}{|\text{tags}(u, r_u)|} \quad (19.18)$$

$$\text{Precision}(\hat{T}(u, r_u)) = \frac{|\text{tags}(u, r_u) \cap \hat{T}(u, r_u)|}{|\hat{T}(u, r_u)|} \quad (19.19)$$

$$F1\text{-measure}(\hat{T}(u, r_u)) = \frac{2 \cdot \text{Recall}(\hat{T}(u, r_u)) \cdot \text{Precision}(\hat{T}(u, r_u))}{\text{Recall}(\hat{T}(u, r_u)) + \text{Precision}(\hat{T}(u, r_u))} \quad (19.20)$$

此外，这两种场景下可以应用于一个在线环境，推荐被实时计算并在用户对资源做标注过程展现给用户。<sup>②</sup>如果用户点击了一个推荐的标签或者使用了推荐(例如，自动补全机制)，可以记录下来。此设置可能是最现实的一个，并为“用户有多喜欢推荐”提供了一个很好的度量。然而，建立起来相当费力，而且需要一个有活跃用户的系统。也有一些用户不点击推荐或使用自动补全，这会影响到评价。

**用户推荐** 对于用户推荐的任务，系统给目标用户建议一个  $n$  个用户的个性化列表，这些用户形成他的近邻。有些系统，像 Last.fm，提供用户与用户之间的连接信息(在 Last.fm，社交关系连接在一起的用户称为邻居)。如果这些信息可用，那么它可以作为用户推荐评估的客观标准。在一些情况下，这样的客观标准不可用(像在 Bibsonomy)，用户推荐评估可以沿着评估用户群体这个思路来进行；最值得一提的是，像[22]中所描述的那样计算群体内物品之间的相似度。特别是在一个社交化书签系统(如 BibSonomy)，对于每个网络资源，它的第一个页面(首页)可以抓取和预处理，创建一个词向量。通过这种方式，任何两个 Web 资源之间的余弦相似性可以计算如下。对于每个测试用户的近邻(即最相似的用户)，可以计算出近邻内的所有 Web 资源对的平均余弦相似度(ACS)。ACS 对应近邻内相似度。此外，所有测试用户的近邻中随机选择一定数目的近邻对，可以计算出所有近邻对之间的平均 Web 资源的相似性。这个度量对应近邻间的相似性。因此，推荐的用户近邻的质量依据其近邻内相似度(越高越好)和近邻间的相似性(越低越好)来判断。

## 19.5 算法比较

在本节中，我们简要地讨论在 19.4 节中介绍的算法的主要优点和缺点。请注意，我们只考虑非基于内容的算法，因为它们可以在一个共同的基础做对比。

在 19.4.1 节中，我们看到为了将基于协同过滤的标准算法应用于大众分类，必须进行一些数据转换。这样的转换导致信息丢失，它会降低推荐质量。基于协同过滤的方法存在另一个众所周知的问题，大型投影矩阵必须保存在内存中，它带来时间/空间消耗，从而影响进行实时推荐的能力。另一个问题是，对每种不同的推荐模式，算法必须最终改变，要求额外的工作来提供多模式推荐。

FolkRank 建立在 PageRank 的基础上，并被证实给出比 CF 明显好的标签推荐[15]。

② 这是一个 2009 年 ECML PKDD 发现挑战赛的任务，见 <http://www.kde.cs.uni-kassel.de/ws/dc09/>。

这种方法也允许模式切换时算法不做变化。此外，同基于协同过滤的算法一样，FolkRank 具有对在线更新的健壮性，因为在一个新的用户、资源或者标签进入系统时，它不需要每次都重新训练。然而，FolkRank 计算昂贵且扩展不便，更适用于实时推荐非必需的系统。

与 FolkRank 一样，张量分解方法直接用于大众分类的三元关系。虽然在学习阶段会有较大消耗(计算量大)，它可以离线执行。完成模型学习后，可以快速作推荐，使这些算法适用于实时推荐。张量分解方法的一个潜在缺点是，仅当不同的推荐问题(即用户/资源/标签)可以通过最小化同样的错误函数来解决时，才能实现简单的模式切换。如果选择 HOSVD，例如，该模型可以通过简单的模式切换来进行多模式推荐，但代价是解决错误的问题：HOSVD 最小化均方误差函数，而社交标签推荐系统与排序更相关。如果你试图面向一个特定推荐模式的错误函数来完美地重建张量，那么，该特定模式的准确度最终会提高，但代价是模式切换更复杂。针对标签推荐问题，图 19.9 显示在 BibSonomy 数据集的一个快照上对上述算法进行了比较[26]。需要注意的是，最好的方法是 RTF，接着是 FolkRank 和 HOSVD。

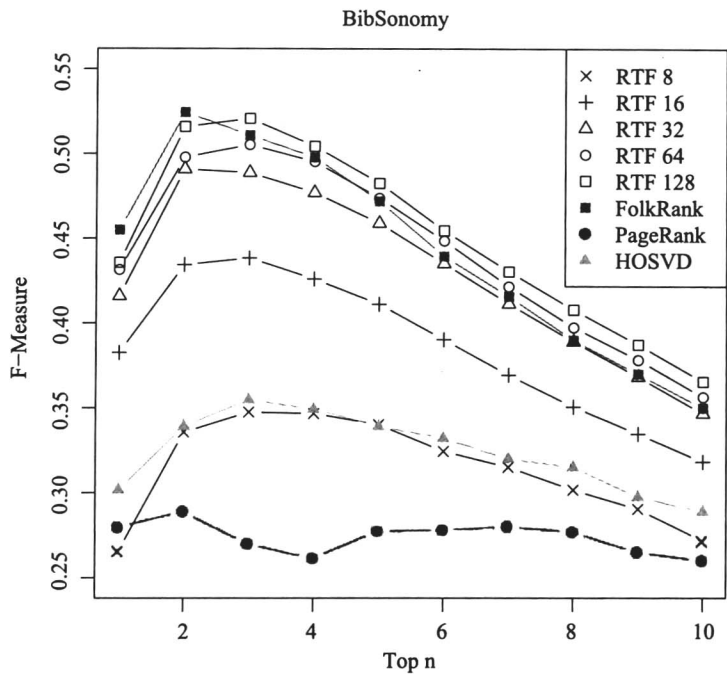


图 19.9 BibSonomy 数据集截图上 Top1、Top2 到 Top10 的 F-算分值得。对比了 FolkRank、PageRank、HOSVD 和弃一法下增加维度数的 RTF

表 19.2 总结了这次讨论。注意，表 19.2 中没有标“X”表示所考虑的算法不能轻易获得相应的特性。

表 19.2 目前算法的优缺点总结

方法	可扩展性	多模式推荐	保存三元关系	在线更新
CF-based				X
Folkrank		X	X	X
HOSVD	X	X	X	
RTF	X		X	

## 19.6 总结与展望

Web 2.0 代表了从 Web 作为信息源到 Web 作为参与平台的转变,用户可以上传内容,行使对这些内容的控制权,在他们使用中增加了应用的价值。Web 2.0 应用中最突出的是社会标签系统,通过将用户的标签/资源暴露给其他用户,促进了标签/资源的共享。由于这些系统日益普及,信息过载等问题迅速成为难题。推荐系统过去被证明非常适合这类难题,因而是解决下一代 Web 信息过载的一个突出手段。在本章中,我们提出:

- 大众分类的数据结构,强调与传统推荐系统所使用的数据结构的差异。
- 可用于社会标签系统中推荐的不同模式。
- 部署在真实社会标签系统上的推荐系统,强调这样做的挑战和要求。
- 获取标签的不同方式,以及它们如何影响推荐算法。
- 算法:
  - 降低数据维数的算法,以应用标准的协同过滤算法。
  - 在大众分类三元关系数据上直接操作的算法。
  - 利用内容资源的算法。
- 评估方案和度量。
- 各个算法的利弊比较。

变换原有大众分类数据的方法虽然可以直接应用于基于标准协同过滤的算法,但是变换不可避免地导致一些信息丢失,这会降低整体的推荐质量。某些方法试图通过对变换产生的不同数据映射进行某种组合来克服这个问题,这增加了额外的自由参数用以控制每个组成部分的影响。一个更自然的解决方案是直接作用于大众分类原有的三元关系上,这就要求研发出新的推荐系统算法如 FolkRank(探究 folksonomy 的超图)或基于张量分解的算法。虽然 FolkRank 以其高品质的预测著称,但也会遇到可扩展性问题,所以一个有趣的研究方向是探讨如何使其具有可扩展性。

用于社会标签推荐系统的张量分解是一个最近突出的研究领域。关于这一主题的研究工作才刚刚开始发现那些方法应当有的优势。一个特别有吸引力的研究方向是具有高推荐精确和模式易切换两个特性的张量分解模型。

正如之前指出的,大众分类通常不包含数值型评分,但最近 GroupLens<sup>⊖</sup>研究小组发布了大众分类的数据集,其中也提供了标签资源的数值型评分。<sup>⊖</sup>这表示有一些研究机会:如何利用资源的评分信息来改善推荐。在这种情况下,所有模式采用一个单一的数据结构(如张量或超图)最终会失败,因为评分仅与“用户—资源”对有关,而与标签无关。对基于内容的方法,也有类似的问题。我们看到,基于内容的方法通常忽略用户信息,但以往的研究表明,结合用户的喜好与资源的内容的混合方法通常会带来更好的推荐系统。在这里,张量或超图表示将会再次失败,因为资源的内容只涉及资源,而不涉及用户或标签。因此,将大众分类的表示方法和资源的内容做某种形式融合的混合方法(hybrid-based methods)将对该领域产生有价值的贡献。

本章未包括一些其他有趣的研究方向,比如,推荐的新颖性和惊喜度[44],即潜在有趣但并不明显的标签、用户和(或)资源;用于推荐的社会智慧建模[40, 29],即显式的好

⊖ <http://www.grouplens.org/>

⊖ 可以从<http://www.grouplens.org/node/73>下载数据集。

友关系的强度和用户之间的互信关系(可能与兴趣、行为的相似度正交),可以建模并用于提升推荐系统的质量。

## 参考文献

1. Rabeeh Abbasi, Marcin Grzegorzczek, and Steffen Staab. Using colors as tags in folksonomies to improve image classification. In *SAMT '08: Poster at Semantics And digital Media Technologies*, 2008.
2. Sergey Brin and Lawrence Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, 30(1-7):107-117, April 1998.
3. Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.*, 22(1):143-177, 2004.
4. P. Drineas and M. W. Mahoney. A randomized algorithm for a tensor-based generalization of the svd. *Linear Algebra and Its Applications*, 420(2-3):553-571.
5. Douglas Eck, Paul Lamere, Thierry Bertin-Mahieux, and Stephen Green. Automatic generation of social tags for music recommendation. In *NIPS'07: Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems*, volume 20, 2007.
6. Claudiu S. Firan, Wolfgang Nejdl, and Raluca Paiu. The benefit of using tag-based profiles. In *LA-WEB '07: Proceedings of the 2007 Latin American Web Conference*, pages 32-41, Washington, DC, USA, 2007. IEEE Computer Society.
7. Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin - Heidelberg, 1999.
8. Gene H. Golub and Charles F. Van Loan. *Matrix computations (3rd ed.)*. Johns Hopkins University Press, Baltimore, MD, USA, 1996.
9. Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5-53, 2004.
10. Thomas Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50-57, New York, NY, USA, 1999. ACM.
11. Andreas Hotho, Dominik Benz, Robert Jäschke, and Beate Krause, editors. *ECML PKDD Discovery Challenge 2008 (RSDC'08)*. Workshop at 18th Europ. Conf. on Machine Learning (ECML'08) / 11th Europ. Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD'08), 2008.
12. Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Information retrieval in folksonomies: Search and ranking. In York Sure and John Domingue, editors, *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, pages 411-426, Heidelberg, June 2006. Springer.
13. Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Trend detection in folksonomies. In *SAMT '06: Proceedings of the first International Conference on Semantics And Digital Media Technology*, volume 4306 of *LNCS*, pages 56-70, Heidelberg, Dec 2006. Springer.
14. Jens Illig, Andreas Hotho, Robert Jäschke, and Gerd Stumme. A comparison of content-based tag recommendations in folksonomy systems. In *KPP '07: Postproceedings of the International Conference on Knowledge Processing in Practice*, 2009 (to appear).
15. Robert Jäschke, Leandro Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. Tag recommendations in social bookmarking systems. *AI Communications*, pages 231-247, 2008.
16. Robert Jäschke, Leandro Balby Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. Tag recommendations in folksonomies. In *PKDD '07: Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, volume 4702 of *Lecture Notes in Computer Science*, pages 506-514, Berlin, Heidelberg, 2007. Springer.
17. Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604-632, 1999.
18. Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455-500, September 2009.
19. Tamara G. Kolda and Jimeng Sun. Scalable tensor decompositions for multi-aspect data mining. In *ICDM '08: Proceedings of the 8th IEEE International Conference on Data Mining*, pages 363-372, December 2008.
20. Leslie Lamport. *LaTeX: A Document Preparation System*. Addison-Wesley, 1986.

21. F. Lehmann and R. Wille. A triadic approach to formal concept analysis. In G. Ellis, R. Levinson, W. Rich, and J. F. Sowa, editors, *Conceptual Structures: Applications, Implementation and Theory*, volume 954 of *Lecture Notes in Computer Science*, pages 32–43. Springer, 1995.
22. Xin Li, Lei Guo, and Yihong Eric Zhao. Tag-based social interest discovery. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 675–684, New York, NY, USA, 2008. ACM.
23. M. Srikanth M. Tatu and T. D'Silva. Tag recommendations using bookmark content. In *ECML/PKDD '08: Proceedings of the ECML PKDD Discovery Challenge at 18th Europ. Conf. on Machine Learning / 11th Europ. Conf. on Principles and Practice of Knowledge Discovery in Databases*, 2008.
24. Leandro Balby Marinho and Lars Schmidt-Thieme. Collaborative tag recommendations. In *GfKL '07: Proceedings of the 31st Annual Conference of the Gesellschaft für Klassifikation (GfKL)*, Freiburg, pages 533–540. Springer, 2007.
25. Oren Patashnik. BibTeXing, 1988. (Included in the BibTeX distribution).
26. Steffen Rendle, Leandro B. Marinho, Alexandros Nanopoulos, and Lars S. Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736. ACM, 2009.
27. Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. GroupLens: an open architecture for collaborative filtering of netnews. In *CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186, New York, NY, USA, 1994. ACM.
28. Lior Rokach and Oded Maimon, Theory and applications of attribute decomposition, IEEE International Conference on Data Mining, IEEE Computer Society Press, pp. 473–480, 2001.
29. Ralf Schenkel, Tom Crecelius, Mouna Kacimi, Thomas Neumann, Josiane Parreira, Marc Spaniol, and Gerhard Weikum. Social wisdom for search and recommendation, June 2008. Accepted for publication.
30. Shilad Sen, Jesse Vig, and John Riedl. Tagommenders: connecting users to items through tags. In *WWW '09: Proceedings of the 18th international conference on World Wide Web*, pages 671–680, New York, NY, USA, 2009. ACM.
31. Andriy Shepitsen, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke. Personalized recommendation in social tagging systems using hierarchical clustering. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 259–266, New York, NY, USA, 2008. ACM.
32. Yang Song, Ziming Zhuang, Huajing Li, Qiankun Zhao, Jia Li, Wang-Chien Lee, and C. Lee Giles. Real-time automatic tag recommendation. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 515–522, New York, NY, USA, 2008. ACM.
33. Sanjay Sood, Sara Owsley, Kristian Hammond, and Larry Birnbaum. Tagassist: Automatic tag suggestion for blog posts. In *ICWSM '07: Proceedings of the International Conference on Weblogs and Social Media*, 2007.
34. Gerd Stumme. A finite state model for on-line analytical processing in triadic contexts. In *ICFCA*, pages 315–328, 2005.
35. Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. Tag recommendations based on tensor dimensionality reduction. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 43–50, New York, NY, USA, 2008. ACM.
36. Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis. *IEEE Transactions on Knowledge and Data Engineering*, 22(2), 2010.
37. Karen H. L. Tso-Sutter, Leandro Balby Marinho, and Lars Schmidt-Thieme. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 1995–1999, New York, NY, USA, 2008. ACM.
38. P. Turney. Empirical evaluation of four tensor decomposition algorithms. *Technical Report (NRC/ERB-1152)*, 2007.
39. Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Commun. ACM*, 51(8):58–67, 2008.
40. Frank E. Walter, Stefano Battiston, and Frank Schweitzer. Personalised and dynamic trust in social networks. In *RecSys '09: Proceedings of the 2009 ACM conference on Recommender systems*, New York, NY, USA, 2009. ACM. to appear.
41. Robert Wetzker, Winfried Umbrath, and Alan Said. A hybrid approach to item recommenda-



- tion in folksonomies. In *ESAIR '09: Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 25–29. ACM, 2009.
42. Wensi Xi, Benyu Zhang, Zheng Chen, Yizhou Lu, Shuicheng Yan, Wei-Ying Ma, and Edward Allan Fox. Link fusion: a unified link analysis framework for multi-type interrelated data objects. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 319–327, New York, NY, USA, 2004. ACM.
  43. Yanfei Xu, Liang Zhang, and Wei Liu. Cubic analysis of social bookmarking for personalized recommendation. pages 733–738. 2006.
  44. Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *WWW '05: Proceedings of the 14th International Conference on World Wide Web*, pages 22–32, New York, NY, USA, 2005. ACM.

## 信任和推荐

Patricia Victor、Martine De Cock 和 Chris Cornelis

**摘要** 推荐技术和信任度是信任增强推荐系统的两大支柱。本章首先讨论信任的一些基本概念，如信任和不信任模型、传播和聚合等。这些概念对于理解信任增强推荐系统技术背后的原理是非常必要的。我们讨论这些技术并且主要集中在信任度和算子在推荐系统中的应用上。其次，我们阐述了推荐算法中使用信任度的好处，总结了信任增强推荐系统中的最新方法。此外，我们详细介绍了三个知名的信任增强推荐系统，并对它们的性能进行了分析与比较。最后的总结部分，我们探讨基于信任的推荐系统领域中技术进展和公开的挑战，例如，推荐系统中的信任关系可视化，如何缓解推荐系统中信任网络的冷启动，推荐过程中不信任的影响，也研究了社交关系中隐含的其他类型关系。

### 20.1 简介

协作、交互与信息共享是 Web2.0[48]时代的驱动力。Web2.0 时代的这些特性也催生了博客(即时分享想法的在线日记或日志)、FOAF(Friend-Of-A-Friend)文件<sup>①</sup>(描述一个人特性的机器可读的文档，包括人与其相互作用物体或人之间的关联)、维基<sup>②</sup>(像 Wikipedia 这样允许人们共同添加并编辑内容的 Web 应用)和社交网站(如 FaceBook<sup>③</sup>、交友网站、车迷论坛等让有着共同兴趣的人们可以交互的虚拟社区)。本章集中于 Web2.0 中的一类特别应用——社交推荐系统。这类推荐系统根据用户个人资料和用户间关系做出预测。如今，在虚拟世界中网络社交关系随处可见，如比较流行的 FaceBook、LinkedIn 和 MSN<sup>④</sup>等。

研究显示相对于根据相似匿名用户的在线系统的推荐，人们更倾向于依赖他们信任朋友的推荐[57]。伴随着社交网络的日益发展和电子商务与推荐系统的结合，人们对信任增强推荐系统产生了越来越多的兴趣。信任增强推荐系统产生的推荐是基于信任网络(社区成员间彼此的信任的一种社交网络)的信息。一个典型的例子是 Golbeck 创办的具备电影评级和评论功能的网站 FilmTrust [16]。该网站要求用户对他们熟悉的电影进行评分，分值范围为 1~10。还有一个例子是电商网站 Epinions，在该网站维护的信任网络中，用户可以根据某人对物品的评点质量来标注他们信任和不信任该成员。用户可以看到信任他的用户，但看不到不信任他的用户。

信任增强推荐系统使用的知识来自于能够生成更个性化推荐信息的信任网络。该网络中用户接受根据信任网络排在较前位置的一些推荐物品，或者甚至是他们信任网中可信朋

Patricia Victor, Chris Cornelis, Dept. of Applied Mathematics and Computer Science, Ghent University, Krijgslaan 281(S9), 9000 Gent, Belgium e-mail: Patricia.Victor, Chris.Cornelis@ugent.be

Martine De Cock, Institute of Technology, University of Washington Tacoma, 1900 Pacific Ave, Tacoma, WA, USA(on leave from Ghent University) e-mail: mdecock@u.washington.edu

翻译: 重庆工商大学-张天水 审核: 李曙光, 胡聪(胡户主), 郑州大学-吴宾

① 见 [www.foaf-project.org](http://www.foaf-project.org)

② 见 [www.wikipedia.org](http://www.wikipedia.org)

③ 见 [www.facebook.com](http://www.facebook.com)

④ 见 [www.linkedin.com](http://www.linkedin.com) 或 [www.msn.com](http://www.msn.com)

友的推荐[7, 16, 46, 61]。这些推荐系统的主要优势是使用了信任传播和聚合算子。该类算子主要原理是估计传递的信任：一种情况是已知用户  $a$  到用户  $b$  的信任值和用户  $b$  到用户  $c$  的信任值来计算用户  $a$  对用户  $c$  的信任值(传播)；另一种情况是把几个信任评估合并成一个最终的信任值(聚合)。传播和聚合是信任度的两个重要的基石，信任度是评估网络中两个陌生用户间的信任程度。

除了信任，用户间(用户都有各自的意图、喜好和观点)自然也可能产生不信任。例如，Epinions 网站首先提出了把用户放进信任名单(基于他们作为一个评论者的质量)，然后增加了不信任名单。不信任名单反映了该用户对某些用户的不信任。信任与不信任名单信息用于对提出的评论进行个性化排序。从研究的角度看，尽管不信任起着非常重要的作用[21, 62, 68]，但该领域研究目前只是处于起步阶段。

推荐技术和信任度构成信任增强推荐系统的两大支柱。由于本书的其他部分对推荐技术进行了详尽的介绍，本章对此只限于介绍必要部分。另外，我们假设大多数读者对信任研究领域不太熟悉。因此，在接下来的章节中，我们会首先讨论基本的信任概念，如信任和不信任模型、传播和聚合。本章讨论的核心就是这些算子在推荐系统中的应用。这些概念对于完全掌握信任增强推荐系统的技术原理是非常必要的，对此我们将在 20.3 节中讨论。作为本章的核心部分，我们将重点放在信任度及其算子的应用上。其次，我们解释了推荐算法中使用信任度的好处，并且简要介绍了信任增强推荐系统中的最新方法。此外，我们详细介绍了三个知名的信任系统，并对它们的性能进行了分析比较。最后，我们在 20.4 节探讨推荐系统领域的一些进展和公开的挑战，如推荐系统中的信任关系可视化，如何缓解推荐系统中信任网络的冷启动，推荐过程中不信任的影响，也研究了社交关系中潜在的其他类型。20.5 节是本章结论。

## 20.2 信任的表示与计算

本节对用户间计算信任进行基本的介绍。在一个社会化信任网络中，这意味着我们需要建立信任模型(20.2.1 节，如何表示和处理信任与不信任)，信任传播算子(20.2.2.1 节，如何通过用户间通路的信息来估计两个用户间的信任值)和信任聚合算子(20.2.2.2 节，如何合并由多条连通路路径生成的信任值)。我们通过经典的和新颖的例子来解释这些概念。

但是我们并未阐述信任网络中的其他概念，如信任更新和信任引导等。因为我们的主要目的是让读者熟悉计算信任领域的主要概念，以便读者能更容易理解 20.3 节中的信任增强推荐技术的原理和细节。

### 20.2.1 信任表示

信任模型可以按照不同的方式进行分类。本章主要集中在两种分类方式，即概率与渐进方法，以及单信任表示方法与信任和不信任的表示方法。表 20.1 针对每种分类方法给出了一些有代表性的参考文献。

概率方法以 0 或者 1 代表用户的信任值，即一个用户要么可信，要么不可信，并用于计算一个用户被信任的概率。这种情况下，一个越高的信任值就意味着用户越可能被信任。在文献[66]中，Zaihrayeu 等提出

表 20.1 信任模型的分类

	信任	信任与不信任
二值	Kamvar 等[28] Richardson 等[55] Zaihrayeu 等[66]	Jøsang 等[25]
渐进	Abdul-Rahman 等[1] Falcone 等[12] Golbeck[15] Massa 等[35]	Victor 等[62] Guha 等[21]

一个扩展的推理框架。这个框架考虑了用户间的信任,也考虑了用户和系统源部件之间的信任。[55]集中于计算应用中的信任,这些应用包含了语义信息,如文献服务器,或者像文献[32]中提到的使用信任系统的社区博客更具抗攻击性。在虚拟网络中,信任是建立在用户间正负面信息传递的基础上的,如 Kamvar 等设计的 P2P 网络 Eigentrust[28],或者 Noh 的基于社会网络反馈的正式模型[44]。文献[25]和[51]使用了一个主观逻辑框架来表示信任值。[25]在 IT 设备中量化和推理信任,而[51]在 P2P 系统中计算用户的信任度。

而渐进的方法是当行为产生一定程度的正向作用的时候估计信任值,即提供的信息一定程度上对或错,并不是完全对或者错(如[1, 11, 15, 21, 35, 59, 68])。在渐进方式下,信任值不是用概率方法解释的:一个用户的信任值越高就意味着信任越高。在这种情况下信任值的排序是一个非常重要的因素。其实在现实生活中也如此,信任通常是一种渐进现象:人们不是信任或者不信任某人,而是某种程度上信任某人。与确定性的数学方法相比较,模糊逻辑[29, 65]是一种非常适合表示这样的模糊区间的自然语言标签。例如,在[59]和[31]中,模糊语言学术语用来在 P2P 网络的用户中表示信任,在社交网络中也如此。在文献[1]中提到了一个经典的例子来表示信任的渐进性。在该文中用 4 个值来表示信任程度:非常信任、信任、不信任、非常不信任。

最近几年,渐进信任方法得到越来越广泛的应用。范围覆盖社会认知模型(如利用模糊认知图[12]),网络上选择好的交流对象管理机制[59],或者普适计算环境(Almenarez 等的 PTM[3]),推荐系统中有用性的表示[15, 35]及为语义网应用设计的模型[68]。

相对于信任模型被逐渐建立,不信任模型却很少被研究。大多数方法都没有研究不信任([31, 32, 43, 55, 66]),或把信任和不信任作为同一连续度量区间下的两端来讨论[1, 19, 59],然而,在用户网络理论中逐渐产生一种观点认为不信任与缺乏信任是等价的[10, 13, 34]。在心理学领域中也存在一些呼声,要求重新认定正负感觉是不可分割的假设[8, 50, 52]。一些研究者甚至认为信任和不信任并不是相对的,在某种情况下可能同时发生[9, 33]。

就我们所知,只有一个概率模型 Jøsang 的主观逻辑[24, 25]中考虑了信任和不信任同时存在的情况[24, 25]。文中假设用户被信任值为  $b$ ,不信任值为  $d$  以及一个不确定值  $u$ 。不确定表示还有部分未知,但是必须满足  $b+d+u=1$ ,排除了不一致性。但其实在大型网络中由于信息源的冲突不一致性是很常见的[60]。

[11, 21, 62, 68]中给出了信任和不信任同时存在的渐进模型。Guha 等用一对  $(t, d)$  来表示信任和不信任,其取值都在  $[0, 1]$ 。而在文献[21]中,他们用  $t-d$  来表示信任值。文献[62]中提到,当不区分信任和不信任时,很多潜在的重要信息都丢失了。例如,一种情况是有 0.2 的信任度,0 的不信任度,而另一种情况是 0.6 的信任度,0.4 的不信任度,最后两者的信任值却都是 0.2,但显然上面两种情况提供的信息差异是很大的。为了处理 Guha 和 Jøsang 方法中的这个问题,Victor 对[11]中的方法进行了扩展,在[14]中信任和不信任值被从两格中抽出来。这样一种两格结构能够解决由于不信任或者知识缺乏引起的信任问题,并且对知识太少或太多造成的问题提供了一些启发,如对立关系、信息量问题[62]。

信任和信任模型用在计算机科学的许多领域和很广泛的应用产品中。在文献[6]中,Artz 和 Gil 进行了全面的梳理,把信任研究分成四个主要方向:利用策略来建立信任模型(访问策略、管理凭证等)、通用模型[12, 68]、信息源信任模型[66]和基于信誉的信任模型。后者包括研究一个用户以前的行为举止[28, 46]和社交网络中计算信任[21, 36]。

20.3 节中的信任增强推荐技术就属于这一类别。

## 20.2.2 信任计算

线上信任网络中,大多数其他用户通常都不认识某个特定用户。当然在某些情况下,我们根据某些信息,可以推出一个陌生用户是否值得信任,以及在多大程度上值得信任。在推荐系统中,如果没有自己认识的用户对他(她)有兴趣的商品打过分,而只有一些陌生的用户(来自信任网络)打过分,那么上面所说的这种推断陌生用户可信度的方法是非常重要的。例如,在 Epinions 网站中,大约每个用户的平均信任用户为 1.7 个, Epinions 网站总共有超过 70 万用户[61]。换句话说,该网中一个用户的信任网络只包含了该社区中的很少一部分用户。因此,能够利用更多的用户的信息生成推荐是非常有用的。

信任度是根据网络上存在的其他用户间的信任关系来估算出一个用户到底应该在多大程度上相信另一个用户。信任度有多种度量方法,[68]对此有一个不错的总结。该文中,Ziegler 和 Lausen 从三个不同纬度对信任度进行分类:成对与成组方式;集中与分布方式;全局与局部方式。第一类是指信任关系被评估的方法,第二类是信任评估被计算的位置,第三类是从网络角度考虑,信任度可以是所有用户及其关系的评估[28, 44, 55],或者是信任网络中部分用户及其关系的评估,这种方式是带有个人偏见的如[15, 21, 35]。20.3 节中的信任技术属于后者。

### 20.2.2.1 传播

信任度通常使用的技术是以信任可以传递为假设的,我们称这些技术为信任传播策略。从图 20.1 中可以看到,如果用户  $a$  信任用户  $b$ (被信任第三方,简称为 TPP), $b$  信任用户  $c$ ,那么假设  $a$  在某种程度上信任  $c$  也是合理的。这种基本的传播策略被称为原子级直接传播,是后面重点研究的传播类型<sup>⊖</sup>。然而,信任不一定总是具有传递性的。例如,如果 Jane 相信 Alice 会给他做一个好看的发型, Alice 相信 John 能给她修好自行车,但并不意味着 Jane 相信 John 会给她修自行车,也不意味着给她做一个好看的发型。

但是,在某些情况下,信任是可以传递的[26]。假如 Jane 是一个镇上的新居民,她想理发。Jane 相信 Alice 能够找到一个好的理发师, Alice 相信 Mariah 是一个好的理发师。因此 Jane 相信 Mariah 是一个好的理发师。这个例子告诉我们必须区分功能性信任(如 Alice 信任 Mariah)和推荐性信任(如 Jane 信任 Alice)[1, 26]。如[26]所述,让信任变得可传递是传播路径可以被看作一个以功能信任范围结束的推荐信任部分传递链。

当仅处理信任时,在概率方式下,乘法常用作标准传播算子[55]。这也是一种渐进方式[3, 15, 21],事实上有多种传播算子,这依赖于应用的目标和场景。下面就是一个实例。

**例 20.1** 假设信任度取值范围为 $[0, 1]$ ,用户  $a$  对用户  $b$  的信任度为 0.5,用户  $b$  对

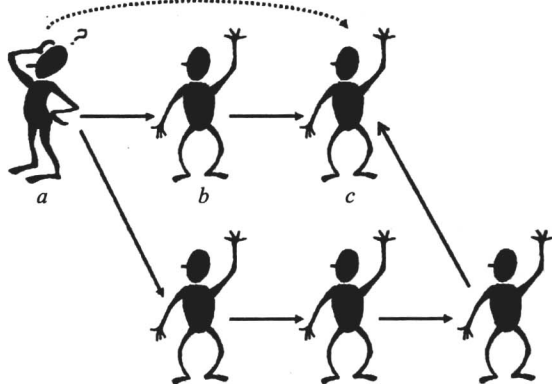


图 20.1 传播实例

<sup>⊖</sup> 如要讨论别的信任传播策略,如分析、信任转置、耦合等可参见[21]。



用户  $c$  的信任度为 0.7, 则概率方式下(标准乘法), 信任传播产生 0.35。但是在模糊逻辑方式下, 最后的信任值依赖于所取算子: 如果传播链只考虑最弱链, 那么传播算子就选择最小信任值, 从而在上面的例子中生成 0.5 信任值。如选择 Łukasiewicz 结合算子, 即  $\max(t_1 - t_2 - 1, 0)$ , 则会产生信任度为 0.2。类似于乘法算子, 这种传播信任值将同时受两个链的影响, 而不仅是最弱链的影响。

另外的一些信任传播技术有: 模糊 if-then 规则[31, 59], 扩散激活理论模型[68], TTP 信任和其感觉间的语义距离[1]。

当然并不是所有的传播路径都有相同的长度。图 20.1 中可见, 从用户  $a$  到用户  $c$  有两条路径。如果我们假设网络中所有的边都表示完全信任。我们会直观感觉到第二条路径的信任度要低于第一条路径。因为两者距离更远。在传播策略中经常会提到信任延时[20]。如 Ziegler 在[68]中的信任延时是通过扩散因子组合起来的。Golbeck 只考虑了最短路径[15], 使用了概率信任而不是渐进信任。Massa 使用基于用户距离的传播信任[35]。

在原子级传播方式下, 如果  $a$  信任  $b$ ,  $b$  信任  $c$ , 则  $a$  也在某种程度上信任  $c$ 。类似地, 如果  $a$  信任  $b$ ,  $b$  不信任  $c$ , 则  $a$  在某种程度上不信任  $c$ 。然而, 当我们在传播链上允许另一条边代表不信任的时候, 情况就变得更复杂了。例如, 如果  $a$  不信任  $b$ ,  $b$  不信任  $c$ , 则  $a$  对  $c$  的信任估计就可能出现几种情况: 一种可能是  $a$  会信任  $c$ , 因为  $a$  可能认为  $b$ (自己不信任的人)不信任的人  $c$  反而是朋友(敌人的敌人就是朋友)。也有可能  $a$  不信任  $c$ 。因为  $a$  认为被某个人不信任的人肯定是不值得信任的。

Guha 等把第二种策略叫作加法不信任传播, 第一种叫作乘法不信任传播[21]。[68]讨论了乘法传播的负效应。但是也总结出由于具备哲学上的可辩论性, 不能被忽略。其他的研究者也提出和第一种策略密切相关的算子。例如, Victor 等的方法使用了模糊逻辑[62], 或者 Jøsang 等提出的亲对立面折扣算子[27]。后一种策略的例子见[21, 27, 62]。

**例 20.2** 和信任传播类似, 不信任传播方法也是用信任和不信任表示亲密关联。假设我们用  $(t, d)$  分别表示信任和不信任, 取值都在  $[0, 1]$ 。(1, 0)表示完全信任, (0, 1)表示完全不信任, (0, 0)表示完全不了解。渐进值如(0.5, 0.2)表示 0.5 的信任, 0.2 的不信任, 也有部分不了解  $1 - 0.5 - 0.2 = 0.3$ 。假设  $a$  对  $b$  的信任范围是  $(t_1, d_1)$ ,  $b$  对  $c$  的信任范围是  $(t_2, d_2)$ , 则  $a$  对  $c$  的信任范围可以通过计算[62]得

$$(t_3, d_3) = (t_1 \times t_2, t_1 \times d_2)$$

传播策略反映了你采纳信任者的意见, 而不会从不信任或者不知道的第三方采纳意见的态度。以下是传播信任值的一些例子。行代表  $a$  对  $b$  的信任值, 列代表  $b$  对  $c$  的信任值, 对应表中也包含了  $a$  对  $c$  的信任值。

	(0.0, 0.0)	(0.0, 1.0)	(1.0, 0.0)	(0.5, 0.2)
(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)
(0.0, 1.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)
(1.0, 0.0)	(0.0, 0.0)	(0.0, 1.0)	(1.0, 0.0)	(0.5, 0.2)
(0.5, 0.2)	(0.0, 0.0)	(0.0, 0.5)	(0.5, 0.0)	(0.25, 0.1)

文献[25]中用这种传播技术来合并信任和不信任。文献[21]中提到的一种传播策略是从信任度中减去不信任度, 则传播信任值为  $t_1 \times (t_2 - d_2)$ 。

**例 20.3** 用户  $a$  对用户  $c$  的信任度  $(t_3, d_3)$  可以通过如下公式计算[62]

$$(t_3, d_3) = (t_1 \times t_2 + d_1 \times d_2 - t_1 \times t_2 \times d_1 \times d_2, t_1 \times d_2 + d_1 \times t_2 - t_1 \times d_2 \times d_1 \times t_2)$$

在这个传播策略中， $t_3$  是  $t_1 \times t_2$  和  $d_1 \times d_2$  的概率和， $d_3$  是  $t_1 \times d_2$  和  $d_1 \times t_2$  的概率和。隐含的假设是一个不信任用户有意给了错误的信息。因此，如果  $a$  信任的人告诉他要信任  $c$ ，则他会信任  $c$ ，如果  $a$  不信任的人告诉他要信  $c$ ，则他会信任  $c$  (即敌人的敌人就是朋友)。从信任度中减去不信任度，则产生值为  $(t_1 - d_1) \times (t_2 - d_2)$ ，[21] 提出了不信任传播策略。下面是一个传播信任值的例子。

	(0.0, 0.0)	(0.0, 1.0)	(1.0, 0.0)	(0.5, 0.2)
(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)	(0.0, 0.0)
(0.0, 1.0)	(0.0, 0.0)	(1.0, 0.0)	(0.0, 0.1)	(0.2, 0.5)
(1.0, 0.0)	(0.0, 0.0)	(0.0, 1.0)	(1.0, 0.0)	(0.5, 0.2)
(0.5, 0.2)	(0.0, 0.0)	(0.2, 0.5)	(0.5, 0.2)	(0.28, 0.2)

所有的这些方法都表明到目前为止人们对不信任的传播机制没有达成共识。人们基于不同的解释提出相应的算子从而产生了不同的结果。这显示在应用中选择合适的传播模式上是一个复杂的问题。

20.2.2.2 聚合

除了传播，信任度量还有聚合策略。毕竟，在一个大型网络中，一个用户的信任值需要通过几条路径获得。在这种情况下，从不同传播路径获得的信任值必须聚合成一个值。实例见图 20.2。

常用的聚合算子有最小值、最大值、加权、平均值或者加权平均等[1, 3, 19, 28, 43, 44, 55]。加权算子让我们可以有机会考虑一些相对更重要的信息源(如 TTP 或传播路径)。

换句话说，加权算子使得聚合过程更灵活。目前为止，信任和不信任聚合还没有得到太多的关注。只有 Jøsang 等为主观逻辑框架提出了三个聚合算子[27]，但是他们在里面假设用户重要性是一样的。

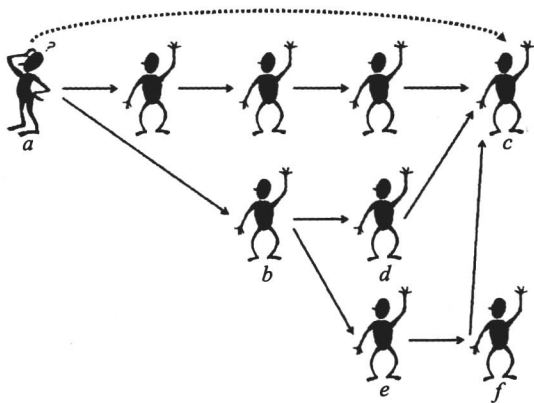


图 20.2 聚合实例

值得注意的是，传播和聚合经常必须在一起使用，而且最后的信任评估依赖于实现的方式。从图 20.2 我们可以看到，用户  $a$  通过用户  $b$  获得对用户  $c$  的信任评估有两种方式。第一种方式是通过两条路径把信任从用户  $a$  传播用户  $c$ ，一条路径是  $a \rightarrow b \rightarrow d \rightarrow c$ ，另一条是  $a \rightarrow b \rightarrow e \rightarrow f \rightarrow c$ ，然后把两条路径的传播信任结果聚合在一起。这种方式下，信任首先被传播，然后被聚合(FPTA)。第二种方式相反，是先聚合后传播。这种方式下，第三方用户  $b$  必须先聚合来自  $d$  和  $e$  的信任，然后把新的信任传递给  $a$ 。可以看到，后者比前者接受了更多的责任。信任可以通过分布式进行计算。而不用用户暴露他们个人的信任或不信任信息。

**例 20.4** 图 20.2 中从用户  $a$  到用户  $c$  有三条不同的路径。假设在最上面一条路径中，除了最后一条边的权重为 0.9，所有其他边的信任权重都为 1，使用乘法作为传播算子，则该路径传播的信任度就为 0.9。现在，假如用户  $a$  对  $b$  的信任度为 1， $b$  对  $d$  的信任度为 0.5， $d$  对  $e$  的信任度为 0.8。这就意味着从用户  $a$  通过用户  $b$  传播到用户  $c$  的两条路径信任度分别

为  $1 \times 0.5 \times 0.4 = 0.2$  和  $1 \times 0.8 \times 0.6 \times 0.7 = 0.34$ 。使用平均值为聚合算子, FTTA 最后产生的信任度为  $(0.9 + 0.2 + 0.34) / 3 = 0.48$ 。另一方面, 假如我们让用户  $b$  首先聚合来自自身的信任网络, 则  $b$  会传播信任度  $(0.2 + 0.34) / 2 = 0.27$  给用户  $a$ 。在 FATP 策略下, 根据图 20.2, 则从用户  $c$  到用户  $a$  传播信任度为  $(0.9 + 0.27) / 2 = 0.59$ 。

## 20.3 信任增强推荐系统

信任增强推荐系统研究的第二个支柱是推荐系统技术。推荐系统常用于精确预测一个用户对某一特别物品的喜欢程度。在许多论文中对这些算法都有介绍 [2, 54]。现在被广泛使用的推荐方法有基于内容的推荐(第 3 章)和基于协同过滤推荐(第 5 章)。基于内容的推荐方法会推荐一些物品, 这些物品与用户曾经喜欢过的物品具有很大的相似性。这种方法往往会限制推荐范围[56], 因此, 这种方法倾向于把推荐限制在和用户购买过或者评分过的商品很近似的范围内。如一个用户  $a$  以前只是租赁了浪漫电影, 那系统就将只推荐相关的浪漫电影, 不会推荐用户可能感兴趣的另外类型电影。使用协同推荐技术可以显著改进推荐系统的推荐效果。这种方式首先识别某一用户和哪些客户具有相似品味(如邻居), 然后基于这些邻居的行为来对该用户的行为做出预测[53]。

20.4 节将讨论经典推荐系统的不足之处, 并且讨论如何通过结合系统中用户信任网络来缓解这些问题。这些先进的基于信任的推荐技术和协同过滤技术是紧密结合的。在这种情况下, 对目标物品的推荐是基于其他用户对这个物品的评分得到的, 而不是对于物品本身属性和内容的分析。Golbeck 在文献[17]中详细全面阐述了信任系统领域, 特别是信任增强推荐系统中的一些经典和新颖的研究工作。

### 20.3.1 动机

尽管推荐方法有了显著的改善, 但是仍然存在一些重要的问题。Massa 和 Avesani 在 [37] 中讨论了一些协同过滤系统的缺点。例如, 用户通常都会给少数的物品打分, 这就使得矩阵非常稀疏(一个推荐系统常会处理数以百万的物品)。例如, Epinions 的一个数据集中有 16 万用户, 2500 万条评分, 150 万条评论[61]。由于数据的稀疏性, 当想在系统中使用协同过滤来识别好的邻居用户的时候就碰到许多困难。这种情况下生成的推荐的质量也得不到保障。要为一个系统的新用户生成一个好的推荐也是很难的, 例如, 冷启动用户因为他们还没有对一定数目的物品进行打分, 不能找到相似的用户群。第三, 由于推荐系统在电子商务中的广泛应用, 促使商家利用推荐系统算法的漏洞, 以便他们的物品能够更频繁地被推荐[67]。例如, 一种普通的“复制身份”攻击手段通过复制目标用户的评分物品给自己创造的恶意用户, 会让系统误认为恶意用户和目标用户的品味很相似。最后, Sinha 和 Swearingen 在 [57, 58] 中提到用户喜欢更透明的系统。因为人们更愿意相信他们信任朋友的推荐而不是根据与他们相似的陌生用户生成的推荐。

现实生活中, 某人有时喜欢一件物品, 为了避免上当, 他通常会更愿意参考他信任的朋友  $a$  的意见。假如  $a$  不了解这个物品, 则  $a$  会问他信任的朋友  $b$ , 直到某个了解这个物品的朋友。信任增强推荐系统就是想仿真这种行为, 就像图 20.3 中描述的这样, 为了得到个性化的推荐, 一旦找到了一条

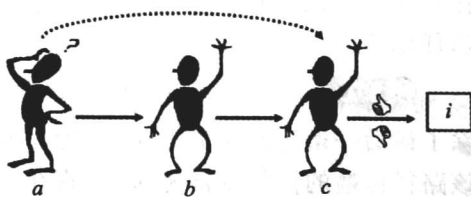


图 20.3 推荐物品

推荐路径, 系统就会通过可得的信任信息(信任传播或者聚合)来合并推荐的判断, 这种方式下, 信任网络就能延伸到越来越多的用户和物品。

在图 20.4 的协同过滤设置中, 用户  $a$  和用户  $b$  给了某些物品  $i_1$  相似的评分, 建立了关联, 用户  $b$  和用户  $c$  建立了类似的关联。结果预测出用户  $a$  对物品  $i_2$  有兴趣。但是在这个场景中, 用户  $a$ (或者  $c$ )和物品  $i_3$  之间并没有关联, 即不能找到物品  $i_3$  是否应该被推荐给用户  $a$  的理由。当在推荐系统中的用户间建立了信任网络的时候, 这种情形就会改变了。图 20.4 的实心线显示用户  $a$  和用户  $b$  之

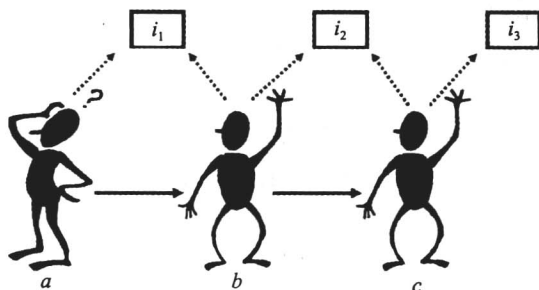


图 20.4 推荐系统中的信任关系

间, 用户  $b$  和用户  $c$  之间存在信任关系。在一个没有信任网络的情形下, 协同过滤系统不能推荐  $i_3$  给用户  $a$ , 而在信任增强情形下: 如果用户  $a$  在一定程度上信任用户  $b$ , 用户  $b$  信任用户  $c$ , 则可以把用户  $a$  的信任传播给用户  $c$ 。如果结果显示用户  $a$  对用户  $c$  有较高的信任度, 则对于用户  $a$  而言,  $i_3$  也许就是一个好的推荐, 并且在被推荐物品中排在较高的位置。这个简单的例子显示通过包含信任关系来扩大推荐系统将有助于解决稀疏问题。甚至, 信任增强系统也会缓解冷启动问题。与评分信息相比, 给同一个账号添加一些信任关系, 则系统生成的推荐会越来越准确[35]。甚至, 信任网络可以产生一个用户可信指标, 使得系统更健壮, 有利于抵御恶意入侵者: 只有在目标用户明确表达了对插入的恶意用户的信任之后, 简单的“复制身份”攻击才会成功。最后, 与“黑箱”方法相比, 信任增强系统的作用(如信任传播)让用户更容易理解。Golbeck 的 FilmTrust 系统[16]要求用户根据他朋友的电影品味来评价他的朋友, 因此使用这些信息就可以生成个性化的预测。

### 20.3.2 进展

所有这些实例都显示在推荐系统的用户中建立信任网络将显著增强推荐性能。有些学者已经在这个领域进行探索[15, 23, 30, 37, 46, 49, 51]。信任增强推荐系统根据信任值采集方式的不同可以分为两类。第一类是信任网络是由用户直接输入, 例如, 明确表示信任, 然后利用该信任网络构建推荐系统, 例子参见[16, 23, 37]。在这种方式下, 我们可以利用网络中的信任传播和聚合生成最终的信任值以满足推荐系统需求。第二类需要用户对他的朋友进行直接评估。信任值是自动生成的, 可以基于用户以前的可信推荐[30, 46], 或者基于相似用户的传递性[49]。

行为学中已经对“信任”有了明确的定义。Mayer 等在文献[40]中定义的信任包含了能力、爱心、正直和倾向等四个主要因素。McAllister 在文献[41]中提出了信任需要区分是基于认知的还是基于情感的。然而, 在推荐研究领域, 信任被看成人际关系的一个总称, 特别是在处理信任值自动计算时。在这种情形下, 信任被用来表示各种概念, 如品味的相似、信誉、能力评估等。

在 20.4 节, 我们将对此作进一步详细讨论。本节集中讨论两类策略(即信任网络挖掘和信任值自动计算), 并对每类策略的代表性工作进行分析。

#### 20.3.2.1 信任网络挖掘

大部分信任增强推荐策略都要求用户表达对其他用户的信任情况。TMoleskiing[7]是一

个山地滑雪社区网站。在这个网站里,他们使用包含信任信息的 FOAF 文件,该信任值的域值范围为 1~9[19]。电子商务网站 Epinions 首先要求用户提交他们网上的信任用户和不信任用户(黑名单),然后以此建立信任网络,网站再根据信任网络对评论进行排序。另一个著名的例子是 Golbeck 的 FilmTrust[16]。FilmTrust 是一个具有电影评分和评论的在线社交网站。在这个网站上,用户被要求对他熟悉的电影喜爱程度进行评分,范围从 1~10 不等。

这些系统都会利用信任网络中关系来确定观点和评分在推荐系统的权重。换句话说,这类算法都会使用信任评价来作为决策过程中的权重。目前有多种赋权方式,在这节,我们会集中讨论两种常用的方式:加权平均和协同过滤自适应机制。我们用一个众所周知的实例来分别阐述它们。

**基于信任的加权平均** 在一个没有信任网络的推荐系统中,如果有一个推荐算法需要求出用户对某一物品  $i$  的喜好程度,则其值为所有对该物品熟悉的用户  $u$  对物品  $i$  评分  $r_{u,i}$  的均值。这种作为基准的推荐策略可以通过计算基于信任的加权平均来改进。设  $t_{a,u}$  表示用户  $a$  对评分者  $u$  的信任程度,算法允许两个个体间信任有差别。现任,被信任程度越高,则其评分权重就越大。在式(20.1)中,  $p_{a,i}$  表示用户  $a$  对物品  $i$  的预测评分,  $R^T$  表示评价物品  $i$  的用户群。

$$p_{a,i} = \frac{\sum_{u \in R^T} t_{a,u} r_{u,i}}{\sum_{u \in R^T} t_{a,u}} \quad (20.1)$$

**TidalTrust** 这个公式是 Golbeck 等的推荐算法的核心[15]。该算法的创新点在于引入了信任估计  $t_{a,u}$ ,他们称为 TidalTrust。在文献[18]中,作者对 TidalTrust 的研究过程进行了全面的介绍。在每一个实验中,他们先是忽略用户  $a$  到  $c$  的直接信任关系,而是考虑  $a$  到  $c$  的其他全部路径。简而言之,通过与这些路径的传播的信任结果与潜在的、隐藏的信任值比较,他们得到结论:1)路径越短产生的信任估计越准确;2)包含越高信任值的路径产生信任结果越好。

从第一条我们知道,只有越短的路径才会产生越好的结果。但是,假如把路径限制到一定长度,则某些情形下,只有一部分用户是可连通的。这种情况下,我们可以设定一个可变路径长度,即把某用户  $u$  ( $u$  是评分者,对物品  $i$  进行了评价)连通的最短长度作为算法路径长度。这样使用广度优先算法搜索的深度也就随着不同用户而变化了。

而第二条,我们可以仅使用最信任用户的路径来计算。然而,每个用户对信任度都有自己不同的理解(一些用户可能常给出相当高的信任度,而其他的用户却不一定)。另外,经常只有几条路径具有一样的高信任值。这也是为什么 Golbeck 等提出要为每条路径增加一个路径强度(例如,这条路径上最小的信任值),并且为每个用户计算所有连接他(她)的路径的所有路径强度中的最大值。这个最大值(max)然后被选择作为最小信任阈值去过滤掉过于小的信任值。

式(20.2)是 TidalTrust 公式。该公式中  $WOT^+(a)$  表示用户  $a$  的信任名单中信任度超过阈值 max 的用户群。这意味着计算每个用户对他的信任者的信任都是加权平均的结果,即只考虑那些他信任对象中信任度不低于 max 的用户。

$$t_{a,u} = \frac{\sum_{v \in WOT^+(a)} t_{a,v} t_{v,u}}{\sum_{v \in WOT^+(a)} t_{a,v}} \quad (20.2)$$



TidalTrust 是一个递归算法,信任度  $t_{a,u}$  是通过递归计算获得的。该算法首先求得用户  $a$  到用户  $u$  的最短路径,然后把该路径中所有的中间用户中的第一条边值作为  $t_{v,u}$  的权重进行加权平均计算得到信任度  $t_{a,u}$ 。用户保证最长路径的深度不会超过当前路径长度。这种算法是渐进信任算法,是一个局部信任度实例。

Golbeck 等通过比较得到基于信任的加权平均与 TidalTrust 相结合的算法不一定比计算平均值或者使用协同过滤更好。但是对于那些给某些特定商品的打分与平均分相差很大的用户,这类方法有非常显著的推荐效果[15, 18]。

**基于信任的协同过滤** Golbeck 的方法是通过加权平均实现的,而信任增强推荐系统的另一类方法是和协同过滤算法紧密相连的。在协同过滤算法中,用户  $a$  对物品  $i$  的评分可以通过用户  $a$  的邻居用户对物品  $i$  的评分得到[53],如式(20.3)所示。

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in R^+} w_{a,u} (r_{u,i} - \bar{r}_u)}{\sum_{u \in R^+} w_{a,u}} \quad (20.3)$$

用户  $a$  对物品  $i$  的评分  $P_{a,i}$  是与用户  $a$  对其他物品的评分的均值  $\bar{r}_a$  和其他用户对物品  $i$  的评分  $r_{u,i}$  相关的。公式考虑了用户  $a$  和用户  $u$  的相似性  $w_{a,u}$  (Pearson 相关系数)[22]。实际上,这里只考虑了那些对物品  $i$  评分的且与用户  $a$  具有正相关性的用户群,记为  $R^+$ 。取代基于 PCC 的权重计算方法,通过信任网络中的用户关系推测出权重(传播和聚合方式)。式(20.4)与式(20.3)相比,用信任度  $t_{a,u}$  取代了 PPC 权重  $w_{a,u}$ 。实例也反映了信任和相似性是相关的,见论文[69]。

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in R^T} t_{a,u} (r_{u,i} - \bar{r}_u)}{\sum_{u \in R^T} t_{a,u}} \quad (20.4)$$

我们把这叫作基于信任的协同过滤。由于权重和 PCC 不是等价的,可能会产生意想不到的结果。当这种情况发生的时候,  $p_{a,i}$  取最近的评分值。

**MoleTrust** 式(20.4)是以 Massa 等的推荐算法为基础的。Massa 在算法中提出了一个新的信任度指标,叫 MoleTrust。该指标计算由两步构成。第一步,去掉信任网络中的环,第二步进行信任计算。由于信任测试集中存在大量的信任传播(如 Epinion 中的大样本测试集),因此首先去掉环会增强效率,这样要获得信任预测值,每个用户只需经过一次。

信任网络变成了一个有向无环图,因此找到用户  $a$  到  $u$  的通路就能够得到  $t_{a,u}$  的信任预测值。首先计算距离为 1 的用户间信任值,然后计算距离为 2 的用户间信任值,以此类推。注意到图的含无环性质,距离为  $x$  的用户的信任值只依赖于已经计算过的用户间距离为  $x-1$  的信任值。

用户间距离为 2 及以上的信任值的计算与 Golbeck 等的算法(式(20.2))相似,但是与广度优先的实现还是有很大差异的。在 TidalTrust 算法中,只有当用户  $u$  位于用户  $a$  到物品  $i$  的最短路径上,它才属于  $WOT^+(a)$ 。

另外,在 MoleTrust 算法中,  $WOT^+(a)$  包含所有的对物品  $i$  进行评分而且用户  $a$  能够直接或通过传播信任关系可以到达的用户。但是并不是所有的用户信任都被计算:在计算开始前,每个用户都必须被分配一个值  $d$  给传播水平参数。这样,只有用户在距离  $d$  内可以到达才会被考虑。MoleTrust 还有一个重要的输入参数是运算中用户的信任阈值(与 TidalTrust 算法中的动态最大阈值不同)。在文献[38]中的实例中用 0.6(取值范围为 0 到

1)作为信任阈值。

与 TidalTrust 算法类似, MoleTrust 算法属于渐进局部信任度方法。Massa 和 Avesani 在实验中说明 MoleTrust 算法比全局信任度算法(如 eBay<sup>⊖</sup>的算法)有更好的信任评估效果,尤其是对那些有争议的用户(这些用户被某一组用户信任而不被另外一组信任)。并且与经典的协同过滤系统相比较, MoleTrust 对冷启动用户的预测效果更好[35, 36]。

Golbeck 和 Massa 的方法是两个利用信任信息的信任增强推荐技术的典型例子。表 20.2 总结了它们的最突出特点。在[23, 63]中也提到了一些在推荐方法中使用信任网络的技术。

表 20.2 两种经典推荐方法的异同比较

算法	TidalTrust	MoleTrust
传播	乘法	乘法
聚合	基于信任的加权平均(20.2)	基于信任的加权平均(20.2)
传播路径最大值	动态(最短路径)	静态(水平参数)
信任阈值	动态(最短 chain 链)	静态
传播中 TTP 需求	$V$ 在 $a$ 到 $u$ 的最短路径上	$V$ 在 $a$ 到 $u$ 的路径上并且距离小于水平参数
评分预测	基于信任的加权平均(20.1)	基于信任的协同过滤(20.4)

### 20.3.2.2 信任自动生成

20.3.2.1 节讨论的算法都要求用户要直接输入信任值。这意味着使用这种算法必须提供一种能够获得这些信息的方法,如 FilmTrus 和 Moleskiing。但很多时候,这种方法是不可行的。这种情况下,自动推断出信任评价而不需要明确的信任信息的方法可能是一个更好的解决方案。这样的系统例子可以在[47]找到。

最常见的方法还是那些基于系统中用户历史评分行为来产生信用机制的方法。一个用户到底应该在多大程度上参与到推荐过程中取决于他在过去准确的推荐。让我们用 O'Donovan 等的经典方法来举例说明[46]。

**用户级和物品级信任** 直觉告诉我们,一个在过去做出了许多高质量推荐的用户,与做得较差的用户相比,前者更值得信任。为了能够选择最值得信任的用户, O'Donovan 引入了两个信任度量:用户级信任和物品级信任。该两个度量分别反映一个用户  $u$  的综合可信度和一个用户  $u$  对某个物品  $i$  的评价的可信度。为了计算用户  $u$  对用户  $a$  的推荐的准确性,这两个信任度都需要。 $r_{a,i}$ 表示用户  $a$  对物品  $i$  的评分,  $p_{a,i}$ 表示用户  $a$  对物品  $i$  评分的预测值。如果通过用户  $u$  预测出  $p_{a,i}$ 在  $r_{a,i}$ 误差范围  $\epsilon$ ,则认为  $p_{a,i}$ 是准确的。

**用户级信任**  $t_u^u$  表示用户  $u$  准确推荐的百分比。注意到这是一个非常广义的信任度;事实上,当用户  $u$  在对某些特定物品进行推荐中完成的比较好时,这是经常发生的。为了实现这个目标 O'Donovan 也提出了更多细粒度的物品级的信任  $t_i^u$ , 这个度量表示物品  $i$  被正确推荐的百分比。因此,在这种自动方式下,信任值不是通过传播和聚合生成的,而是基于过去的评分。注意到 O'Donovan 的方法是全局信任度。值获得的方式可以被认为是概率的。

**基于信任的过滤** 与别的信任增强技术相似,通过信任度量获得的值可以在推荐过程中作为权重。像 Massa 一样, O'Donovan 等的方法集中于协同过滤中加入信任。文献[46]中做出了几种尝试。如把 PCC 信息和获得的信任值合并,这种方式的一个替代方式

是使用信任值作为过滤器, 这样在推荐过程中只有最值得信任的邻居参与进来。这种策略被叫作基于信任的过滤, 见式(20.5),  $w_{a,u}$  表示 PCC,  $R^{T+} = R^T \cap R^+$ .

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in R^{T+}} w_{a,u} (r_{u,i} - \bar{r}_u)}{\sum_{u \in R^{T+}} w_{a,u}} \quad (20.5)$$

另外, 只有物品或用户级别的信任高于某特定阈值并且与  $a$  用户正相关的用户才会被考虑。

在文献[46]中, O'Donovan 和 Smyth 展示了基于信任的过滤方法比协同过滤在平均错误率上取得更好的结果。而且, 在几乎 70% 的预测实例上, 基于用户级别信任的算法比协同过滤方法产生更少的错误。

在自动生成信任的方法中, O'Donovan 的方法是一个代表性例子。可以在文献[30]中找到一个相似的方法, 用一个实用度量方法来代替一个二值正确性函数。

### 20.3.3 实验比较

那么在实际应用中, 上面讨论的哪一种方法是最佳的呢? 目前为止, 基于信任的推荐领域中的研究者都是使用自己的数据集和应用来介绍和评估他们提出的新算法, 而没有在同一个数据集或者应用中与其他的信任增强算法作比较。因此, 本节接下来将在同一个数据集上对前面提到的几种信任增强算法进行性能上的比较。我们将集中比较 Golbeck 的基于信任的加权平均算法 TidalTrust、Massa 的基于信任的协同过滤算法 MoleTrust 和 O'Donovan 的基于信任的过滤算法(式(20.5))。由于我们的目的是比较在同一数据集中的三种算法, 为了研究信任传播的影响, 我们没有选择完成 O'Donovan 的自动信任生成策略, 而是挖掘同样的信任网络作为另外的两种策略。尽管 O'Donovan 等没有在他们的实例中使用信任传播算子, 这样做当然是可能的。由于在式(20.5)中没有明确使用信任值, 我们只需要指出信任传播是如何扩大  $R^{T+}$  的。

#### 20.3.3.1 数据集

我们的实验中使用的数据集来自 Epinions.Com。该网站中用户对消费的物品进行评论并对物品和评论进行评分。在信任增强推荐系统实验中常用两个 Epinions 数据集。第一个数据集是 Massa 和 Bhattacharje[39]用了 5 个星期的爬虫收集到的, 包含 49290 个用户, 139738 个物品。评分范围为 1~5。第二个数据集是由 Guha 等[21]收集的: 这个数据集包含 163634 个用户, 1560144 条评论, 25170637 个评分。对评论的评分范围从“没有帮助(1分)”到“最大帮助(5分)”。这个数据集不包含关于物品和物品评分的任何信息, 而是用评论和评论评分代替了。换句话说, 这个数据集是对评论推荐系统进行讨论和评估。因此, 在与 Guha 相关的数据集中, 一个物品表示对一个消费商品的评论, 而对爬虫数据集, 一个物品表示一个消费商品。

对于随机的和那些有争议的物品, 我们的实验关注系统生成的推荐/预测的数量以及预测的错误率。那些有争议的物品是对于推荐系统最有挑战的物品, 因为如果一个物品获得的分数差异很大, 那么反映了对该物品的意见不一致。更多的情况是, 若目标商品是有争议的, 例如, 当一个商品拥有数目相当的热心支持者和积极的反对者, 那么被推荐的物品需要真正的个性化。在文献[63]中 Victor 等解释了为什么用经典的标准差方法来表示数据集中有争议的物品是不够的, 并且提出一个新的度量来表示一个物品的争议程度。他们

的方法分别在 Guha 和 Massa 的数据集中找到了 1416 和 266 个争议物品。关于争议性的计算,更多细节可参见文献[63]。为了对比在争议的物品(CI)和一般商品上的性能,我们分别在两个数据集中随机选择 1416 个和 266 个畅销物品(RI,类似争议商品,至少被评估过 20 次),其中也展示了不同方法在平均覆盖率和准确度上的结果。

Epinions 允许用户针对其他用户的评论质量对用户打分。这样不仅可以评分,还可以进行信任和不信任评估。事实上两个数据集都包含了来自用户的明确信任信息,这使得这两个数据集非常适合信任增强推荐系统的研究。用户可以评估其他用户,以确定把他们放进 WOT(即评论和评分都被发现是有价值的评论者列表<sup>⊖</sup>)或者不信任列表(评论被发现是攻击性、不准确或低质量,也就是不可信的作者列表)。在 Guha 的数据集中,Epinions 信任图由 114222 个用户和 717 129 个信任关系(不包含自己信任)构成。Massa 的数据集由 49 288 个用户以及 487 003 个信任评价构成。

由于数据集只包含二值信任,所以在我们的实验中,式(20.1)、式(20.4)和式(20.5)的  $t_{a,u}$  中只考虑值 0(完全不信任)和 1(完全信任)。这个限制导致了一些基于信任算法的改变,如式(20.1)简化成了平均。为了简单,我们在这篇论文中只考虑一步传播。这意味着式(20.4)和式(20.5)中,我们只考虑长度为 1 和 2 的传播链,而在式(20.2)中由于不能得到更短的传播链,我们只考虑长度为 2 的传播链。这两步简化限制了我们的实验比较,因为我们不能完全按照设计的算法对它们进行对比分析。

### 20.3.3.2 覆盖率

覆盖率是指由预测产生的用户-物品配对数。一个传统的评价推荐系统覆盖率的方法是留一法,即隐藏一个评分然后预测这个隐藏值。覆盖率的算法是通过  $p_{a,i}$  除以留一法产生的数目来实现的(即数据集中有效的评分数)。式(20.3)中,我们称  $p_{a,i}$  是可计算的,因为至少有一个用户  $u$  可以被计算,而在式(20.1)和式(20.4)中,  $p_{a,i}$  表示至少有一个用户  $u$  的  $t_{a,u}$  是可计算的。最后,式(20.5)中,若至少有一个用户  $u$  的 PCC 可计算和  $t_{a,u}=1$ ,则是可预测的。

表 20.3 显示了 Guha 和 Massa 数据集中的争议物品(CI)和随机选择物品(RI)的覆盖率。前四行包含四个基准策略(B1)-(B4)。第一个基准策略 B1 是推荐系统总是做出 5 分(总分 5 分)的预测,而 5 分是 Epinions 上物品最频繁的得分。第二个基准策略使用物品的平均得分(B2),第三个基准策略使用用户的平均得分(B3)。后者使用那些评分差异度小的用户的评分。第四个基准策略返回一个 1~5 随机分(B4)。

总的来说,策略 B1、B2、B4 对有争议的和随机选定的物品都获得了 100% 覆盖率: B1 和 B4 没有使用别的附加信任信息,因为我们的实验中,这些物品至少被评分 20 次,所以 B2 总是可以计算的。而 B3 在那些用户只对一个物品打分的情况下,不能产生平均评分,因此不能产生预测。

对表 20.3 中的其他算法,第一列表示上文给出的推荐公式。信任增强算法中,我们分别使用了传播信任信息的(上面几行)和未使用传播信任信息的(下面几行)实验。我们只考虑一步传播: P1 和 P4,我们分别使用了 TidalTrust 和 ModelTrust<sup>⊖</sup>中的传播策略。而在 P5 中,如果一个用户属于用户  $a$  的信任网络或者被用户  $a$  的信任网络的成员信任,则我们把该用户包含进 RT。

⊖ [www.epinions.com/help/faq/](http://www.epinions.com/help/faq/)

⊖ 注意,我们包含了 Massa 等针对二元信任环境的基于水平的策略[35]。

表 20.3 基于信任的推荐算法性能比较

		Guha 等的数据集						Massa 等的数据集					
		争议物品			随机选择物品			争议物品			随机选择物品		
算法		% COV	MAE	RMSE	% COV	MAE	RMSE	% COV	MAE	RMSE	% COV	MAE	RMSE
(B1)	基准算法：分数 5	100	1.45	1.96	100	0.16	0.51	100	1.94	2.46	100	1.05	1.62
(B2)	基准算法：物品平均分	100	1.25	1.34	100	0.18	0.40	100	1.35	1.51	100	0.82	1.06
(B3)	基准算法：用户平均分	99	1.12	1.58	100	0.36	0.50	98	1.43	1.78	99	0.95	1.22
(B4)	基准算法：随机分数	100	1.61	2.02	100	1.92	2.37	100	1.66	2.08	100	1.68	2.10
(20.3)	协同过滤	94	0.96	1.13	98	0.19	0.38	81	1.34	1.58	79	0.84	1.12
(20.1)	基于信任的加权平均	63	0.86	1.20	89	0.13	0.35	41	1.33	1.70	34	0.87	1.24
(20.4)	基于信任的协同过滤	63	0.87	1.16	89	0.17	0.35	40	1.32	1.65	34	0.86	1.19
(20.5)	基于信任的过滤	60	0.86	1.16	86	0.16	0.36	25	1.35	1.71	22	0.85	1.18
(P1)	传播基于信任的加权平均	88	0.91	1.22	97	0.15	0.38	76	1.37	1.69	72	0.90	1.23
(P4)	传播基于信任的协同过滤	88	0.99	1.16	97	0.19	0.37	76	1.32	1.56	72	0.84	1.12
(P5)	传播基于信任的过滤	84	0.94	1.13	96	0.18	0.36	57	1.36	1.64	53	0.86	1.16

如果没有信任传播，与其他算法相比，协同过滤算法的覆盖率无疑是最好的。这是由于通常 PCC 信息比直接的信任信息更容易得到：用户  $a$  有正相关的用户比  $a$  信任网络中的用户更多。另一方面，使用了 PCC 权加权的基于信任的过滤式(20.5)算法是最苛刻的策略，因为它要求用户  $a$  的信任网络中的用户和  $a$  一样对其他物品打过二次分(否则 PCC 就不能被计算)。TidalTrust 式(20.1)的覆盖率比 ModelTrust 式(20.4)的覆盖率高一点，这是由于后者只能对那些至少对两个物品打过的用户生成推荐预测(否则用户的平均得分就不能被计算)。

当加入信任传递后，这些方法依据覆盖率排序的顺序依然保持不变。但是值得注意的是，协同过滤方法由于考虑了传播，在 Guha 数据集中对于争议和随机选择物品，覆盖率分别增加了 25%和 10%，而在 Massa 的数据集中超过了 30%。

Guha 的数据集中争议物品覆盖率显著低于随机选择物品。这是由于该数据集中争议物品比随机选择物品较少获得评分机会，这导致这些物品不仅在留一法中较少，而且较少机会获得一个拥有正面 PCC 的用户评分信任值。这也解释了复杂的推荐策略具有较低的推荐率。而 Massa 的数据集中就得不到类似的结果，CI 比 RI 获得了更多的评分(21 131 : 12 741)。这解释了随机选择物品集数据集算法性能在覆盖率中有点差的结果。

这也意味着 Massa 的数据集的覆盖率要显著低于 Guha 的数据集。式(20.1)、式(20.4)、式(20.5)获得的覆盖率至少要低 20%。相对于 Massa 数据集而言，Guha 数据集中用户给更多的物品评分，这就导致给同一个物品评分的用户更少，即邻居用户更少。

20.3.3.3 准确率

推荐系统的准确率通常根据留一法进行评价，更多的是由隐藏评分和预测评分的差来



决定的。我们使用常用的两个度量指标：平均绝对误差(MAE)和平均平方根误差(RMAE)[22]。前者同样对待每一个误差而后者强调较大的误差。由于评论和物品的评分同样都采用5分对待每一个误差值。MAE和RMSE中的极差可以达到0和4。RMSE的一个细微的改进在推荐系统中都是非常有价值的。如NetFlix竞赛<sup>①</sup>主办方提供了100万美元给RMSE减低了10%的获胜者。

表20.3中争议物品的MAE和RMSE都比随机选择物品要高。也就是说，相对于随机选择物品，争议物品产生更好的推荐结果更难。这个结论对于所有的算法都适用，但是基准策略的几个算法更明显地体现出来(除了B4)。然而在Massa数据集中，所有的算法变化趋势都差不多，在Guha数据集中B1和B2要对争议物品产生预测更困难；而对于随机选择物品，它们可以和协同过滤和基于信任算法相媲美。在争议物品集中，它们的MAE和RMSE增加了1分。

同时也需注意，不管是争议物品还是随机物品，在Massa中产生好的推荐比Guha中更难。这是由于Massa数据集中物品自身就具有更多争议性。

我们注意到，不考虑传播性，使用非基准线中方法计算争议物品的MA时，信任增强方法都比协同过滤产生更好的结果<sup>②</sup>(在Massa数据集中使用基于信任的协同过滤时只有一个例外)，这与文献[15, 36]中的结果是一致的。这可以归结到准确率和覆盖率的关系：覆盖率增强意味着准确率下降，反之亦然。很显然当考虑了信任传播算子的时候，信任增强算法与协同过滤算法的覆盖率很接近。MAE也是这样。

而RMSE却并不如此。在争议物品集中，信任增强算法的RMSE通常比协同过滤算法要高，而在随机物品集中却不一定？较高的RMSE意味着较大的预测误差。对争议物品而言，这可能是由于 $R^T$ 集中对指定物品的打分人数太少(如只有一个用户)，特别是比 $R^+$ 更小。这个假设也被如下事实支持：由于信任传播网络中RMSE的增强比MAE慢。另外，传播算法经常比对应的非传播算法获得更低的RMSE，参见Massa数据集中的争议物品结果。

#### 20.3.3.4 结论

实验中的两个Epinions数据集都有各自的特点，却都获得了相同的结论。对随机物品，智能推荐策略(如协同过滤、基于信任的算法)和基线方法表现差不多。而对争议物品，智能推荐算法表现要好得多。尽管还有提升性能的余地，但是信任增强推荐系统相对其他方法没有明显的优势；较高的RMSE值以及在Massa数据集的CI上，基于信任的方法并不会产生比协同过滤更好的结果都是事实。这些发现要求在改进算法和信任算法有效性的适应性方面有更深的研究(例如，对于冷启动用户Massa等的结果是什么)。

通过覆盖率和精确率对三种最佳信任增强策略(分别由Golbeck等、Massa等、O'Donovan等提出)进行评估的结果没有显示哪种策略具有明显的优势。基于信任的协同过滤策略在Massa的数据集上得分最高，而基于信任的加权平均和基于信任的过滤在Guha的数据集上精确度最高。根据信任传播获得的结果也类似。

由于这两个数据集包含了评分信息和信任信息，它们在信任增强推荐测试中很常用。然而，这两个数据集有一个缺点：在Epinions数据集中的信任值是二值的。由于许多现有的信任方法都是基于信任值是连续的假设之下，这使得我们要想探讨本章讨论的各种算法是不行的。遗憾的是，目前还没有公开的信任值是连续值的数据集。

① <http://www.netflixprize.com/>

② 注意，所有在Guha数据集上的MAE提升在统计上都是显著的( $p < 0.000$ )。

## 20.4 进展和开放性挑战

在前面几节,我们介绍了信任建模、信任度量、信任增强推荐系统的基础知识。本节我们想给读者提出在基于信任的推荐系统领域中未来可能的新的研究方向。这并不意味着是一个全面的介绍,而只是有选择地介绍该领域的一些进展。我们将特别简短地讨论以下问题:缓解基于信任的冷启动,信任增强推进系统的可视化,基于信任研究的理论基础,把不信任引入推荐过程。

Massa 和 Avesani 的研究显示通过在推荐系统中引入用户信任网络能有效缓解常碰到的冷启动问题。他们证明了对一个新用户提出一些信任问题让其回答(与对物品评分比较),对于其在系统中获得一个好的推荐是非常有好处的[35]。然而,Victor 等研究显示通常是冷启动的用户(仅对很少一部分物品评分)在信任网络中也是冷启动用户[61]。因此,一定要鼓励新用户和别的用户建立联系来尽可能地拓展信任网络,但是选择谁来建立联系经常是一个难题。在业已交付的推荐中,指导这些新来者如何在早期阶段中建立联系是非常重要的。在文献[61]中,这个问题是通过推荐系统网络中的三种关键角色来解决的:资深评分用户、专家、连接者。作者指出,对一个冷启动用户,连接到一个标志性的关键人物比随机选择一个用户在推荐系统中的覆盖率和精确率上都更具优势。

注意到这个连接引导问题牵扯到更广泛的信任引导问题,即如何在网络中建立一个初始信任关系的问题。O'Donovan 用了一种非常特别的方法也解决这一冷启动问题,他的方法叫 PeerChooser,这是一个新的对基于信任的协同过滤系统进行可视化的程序。具体来说,PeerChooser 不仅能够对来自传统的相似度方法 PCC 的信息进行可视化,而且可以对由评分数据(我们前文提到了 O'Donovan 在[46]中提出的信息和物品数据)生成的信任空间信息进行可视化。这个软件的主要特点之一是,它在推荐时能够从用户那里获得信任信息,这是通过在交互界面上通过移动用户图标来实现的。这样用户不仅可以表达他们的情绪和喜好等,而且更愿意主动。

当用户间信任关系不能明确给出的时候,也有一些其他的方法来建立信任关系。社交数据是可以作为参考信息来源的,如在线社交和商业往来(如 Facebook 和 LinkedIn)、Email 通信、信誉系统等。在推荐系统的研究中,这些数据源都是作为信任关系来讨论的,尽管他们对应不同的概念:行为学理论区分成了同质性与认知相似性,社交资本(信誉、意见领袖等)、紧密程度(关系持续时间、互动频率)和信任[40, 41]。这些社交数据源可以用于(信任增强)推荐系统,但是研究并未指出到底哪种数据源是最有用的[4]。是否这些数据源都可能会像本章讨论的经典的基于信任的推荐系统一样提供类似的结果。在文献[5]中,Arazy 等提出这个问题,并争论称社交推荐系统的设计应该基于某种理论,而不应该为了某种目的而特别设计,尽管当前的算法经常采用这样的设计方式。

另一个近来的一个研究方向是基于信任推荐系统中的另一个属性,即不信任。事实上,在信任建模领域中很少有关于不信任的研究,在推荐领域中则更少。这是几方面的原因造成的,最重要的原因是很少能得到不信任信息的数据源,另外人们也不知道不信任如何传播,在推荐中如何使用它形成一致意见。文献[64]第一次在推荐过程中对使用不信任进行了实验评价。一共研究了三种不信任策略,第一种把不信任作为逆向偏差指示器,而第二种把不信任用来过滤邻居用户,第三种把不信任作为信任网络的调试器。前两种策略是基于在协同过滤系统中信任可以用于选择相似用户(近邻)的原理,后者是该领域中的不同研究者提出来的[20, 68]。结果显示第一种策略不值得考虑,而用不信任去过滤或者调试看起来更有希望。但是很显然,在得到更精确的结果之前,这个新生的研究领域还有很多工作要做。

## 20.5 总结

本章对信任建模领域的研究做了介绍,阐述了信任网络是如何增强传统推荐系统性能的。我们讨论了几种效果最好的信任增强推荐策略,并利用 Epinions.com 中的两个数据集做实验来测试这几种方法的性能。根据覆盖率和准确率并不能明确区分哪种算法具有明显的优势,但是也显示每种算法都有自己的优点。

拥有用户信任网络的推荐系统可以利用信任聚合策略来增强推荐性能。而在那些没有确切信任关系的推荐系统中,能够自动计算信任度的方法目前来看是最理想的解决方案。当然,如果没有适合的数据集或者应用来测试,这些策略也不会被创造出来。

事实上,缺乏公开有效适用的测试数据集是信任增强推荐系统研究面临的最大困难之一。所以我们在不久的将来每一个这个领域的研究员都可以获得更多这样的数据和应用。我们也有足够的信心相信这个领域将吸引越来越多的研究者,并且促进基于信任推荐系统研究领域的蓬勃发展。

## 参考文献

1. Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Proc. of the 33rd Hawaii International Conference on System Sciences, pp. 1769-1777 (2000)
2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17, 734-749 (2005)
3. Almenárez, F., Marín, A., Campo, C., García, C.: PTM: A pervasive trust management model for dynamic open environments. In: Proc. of the First Workshop on Pervasive Security, Privacy and Trust, in conjunction with Mobiquitous (2004)
4. Arazy, O., Elsane, I., Shapira, B., Kumar, N.: Social relationships in recommender systems. In: Proc. of the 17th Workshop on Information Technologies & Systems (2007)
5. Arazy, O., Kumar, N., Shapira, B.: Improving social recommender systems. *IT Professional* May/June, 31-37 (2009)
6. Artz, D., Gil, Y.: A survey of trust in computer science and the semantic web. *Journal of Web Semantics* 5, 58-71 (2007)
7. Avesani, P., Massa, P., Tiella, R.: Moleskiing.it: a trust-aware recommender system for ski mountaineering. *International Journal for Infonomics* (2005)
8. Cacioppo, J., Berntson, G.: Relationship between attitudes and evaluative space: a critical review, with emphasis on the separability of positive and negative substrates. *Psychological Bulletin* 115, 401-423 (1994)
9. Constantinople, A.: An eriksonian measure of personality development in college students. *Development Psychology* 1, 357-372 (1969)
10. Cofta, P.: Distrust. In: Proc. of the International Conference on Electronic Commerce, pp. 250-258 (2006)
11. De Cock, M., Pinheiro da Silva, P.: A many-valued representation and propagation of trust and distrust. In: Bloch, I., Petrosino, A., Tettamanzi, A. (eds.) *Lecture Notes in Computer Science* 3849, pp. 108-113 (2006)
12. Falcone, R., Pezzulo, G., Castelfranchi, C.: A fuzzy approach to a belief-based trust computation. In: Eder, J., Haav, H-M., Kalja, A., Penjam, J. (eds.) *Lecture Notes in Artificial Intelligence* 2631, pp. 73-86 (2003)
13. Gans, G., Jarke, M., Kethers, S., Lakemeyer, G.: Modeling the impact of trust and distrust in agent networks. In: Proc. of the Third Workshop on Agent-oriented Information Systems, pp. 45-58 (2001)
14. Ginsberg, M.: Multi-valued logics: A uniform approach to reasoning in artificial intelligence. *Computational Intelligence* 4, 265-316 (1988)
15. Golbeck, J.: Computing and applying trust in web-based social networks. PhD thesis (2005)
16. Golbeck, J.: Generating predictive movie ratings from trust in social networks. In: Stølen, K., Winsborough, W.H., Martinelli, F., Massacci, F. (eds.) *Lecture Notes in Computer Science* 3986, pp. 93-104 (2006)

17. Golbeck, J.: *Computing with Social Trust*. Springer, London (2009)
18. Golbeck, J., Mannes, A.: Using trust and provenance for content filtering on the semantic web. In: *Proc. of the WWW06 Models of Trust for the Web Workshop* (2006)
19. Golbeck, J., Parsia, B., Hendler, J.: Trust networks on the semantic web. In: Klusch, M., Omicini, A., Ossowski, S., Laamanen, H. (eds.) *Lecture Notes in Artificial Intelligence* 2782, pp. 238-249 (2003)
20. Guha, R.: Open rating systems. Technical report, Stanford Knowledge Systems Laboratory (2003)
21. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: *Proc. of the World Wide Web Conference*, pp. 403-412 (2004)
22. Herlocker, J., Konstan, J., Terveen, L., Riedl, J.: Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* **22**, 5-53 (2004)
23. Hess, C., Schiedler, C.: Trust-based recommendations for documents. *AI Communications* **21**, 145-153 (2008)
24. Jøsang, A.: A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **9**, 279-311 (2001).
25. Jøsang, A., Knapkog, S.: A metric for trusted systems. In: *Proc. of the National Computer Security Conference*, pp. 16-29 (1998)
26. Jøsang, A., Gray, E., Kinader, M.: Simplification and analysis of transitive trust networks. *Web Intelligence and Agent Systems* **4**, 139-161 (2006)
27. Jøsang, A., Marsh, S., Pope, S.: Exploring different types of trust propagation. In: Stølen, K., Winsborough, W.H., Martinelli, F., Massacci, F. (eds.) *Lecture Notes in Computer Science* 3986, pp. 179-192 (2006)
28. Kamvar, S., Schlosser, M., Garcia-Molina, H.: The eigentrust algorithm for reputation management in P2P networks. In: *Proc. of the World Wide Web Conference*, pp. 640-651 (2003)
29. Klir, G., Yuan, B.: *Fuzzy sets and systems: theory and applications*. Prentice Hall PTR, New Jersey (1995)
30. Lathia, N., Hailes, S., Capra, L.: Trust-based collaborative filtering. In: Karabulut, Y., Mitchell, J., Herrmann, P., Damsgaard Jensen, C. (eds.) *IFIP International Federation for Information Processing* **263**, 119-134 (2008)
31. Lesani, M., Bagheri, S.: Applying and inferring fuzzy trust in semantic web social networks. In: Kodé, M.T., Lemire, D. (eds.) *Semantic Web and Beyond 2*, pp. 23-43 (2006)
32. Levien, R.: Attack-resistant trust metrics. In: Golbeck, J. (ed.) *Computing With Social Trust*, pp. 121-132 (2009)
33. Lewicki, R., McAllister, D., Bies, R.: Trust and distrust: new relationships and realities. *Academy of Management Review* **23**, 438-458 (1998)
34. Marsh, S., Briggs, P.: Examining trust, forgiveness and regret as computational concepts. In: Golbeck, J. (ed.) *Computing With Social Trust*, pp. 9-43 (2009)
35. Massa, P., Avesani, P.: Trust-aware collaborative filtering for recommender systems. In: *Proc. of the Federated International Conference On The Move to Meaningful Internet*, pp. 492-508 (2004)
36. Massa, P., Avesani, P.: Trust-aware recommender systems. In: *Proc. of ACM Recommender Systems*, pp. 17-24 (2007)
37. Massa, P., Avesani, P.: Trust metrics in recommender systems. In: Golbeck, J. (ed.) *Computing with Social Trust*, pp. 259-285 (2009)
38. Massa, P., Avesani, P.: Trust metrics on controversial users: balancing between tyranny of the majority and echo chambers. *International Journal on Semantic Web and Information Systems* **3**, 39-64 (2007)
39. Massa, P., Bhattacharjee, B.: Using trust in recommender systems: an experimental analysis. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) *Lecture Notes in Computer Science* 2995, pp. 221-235 (2004)
40. Mayer, R., Davis, J., Schoorman, D.: An integrative model of organizational trust. *The Academy of Management Review* **20**, 709-734 (1995)
41. McAllister, D.: Affect- and cognition-based trust as foundations for interpersonal cooperation in organizations. *The Academy of Management Journal* **38**, 24-59 (1995)
42. Moskovitch, R., Elovici, Y., Rokach, L.: Detection of unknown computer worms based on behavioral classification of the host, *Computational Statistics and Data Analysis*, 52(9):4544-4566 (2008)
43. Mui, L., Mohtashemi, M., Halberstadt, A.: A computational model of trust and reputation. In: *Proc. of the 35th Hawaii International Conference on System Sciences*, pp. 2431-2439 (2002)
44. Noh, S.: Calculating trust using aggregation rules in social networks. In: Xiao, B., Yang,

- L., Ma, J., Muller-Schloer, C., Hua, Y. (eds.) *Lecture Notes in Computer Science* 4610, pp. 361-371 (2007)
45. O'Donovan, J.: Capturing trust in social web applications. In: Golbeck, J. (ed.) *Computing With Social Trust*, pp. 213-257 (2009)
46. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: *Proc. of the 10th International Conference on Intelligent User Interfaces*, pp. 167-174 (2005)
47. O'Donovan, J., Smyth, B.: Mining trust values from recommendation errors. *International Journal on Artificial Intelligence Tools* 15, 945-962 (2006)
48. O'Reilly, T.: What is web 2.0. Available at <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html> (2005)
49. Papagelis, M., Plexousakis, D., Kutsuras, T.: Alleviating the sparsity problem of collaborative filtering using trust inferences. In: Herrmann, P., Issarny, V., Shiu, S. (eds.) *Lecture Notes in Computer Science* 3477, pp. 224-239 (2005)
50. Petty, R., Wegener, D., Fabrigar, L.: Attitudes and attitude change. *Annual Review of Psychology* 48, 609-647 (1997)
51. Pitsilis, G., Marshall, L.: A trust-enabled P2P recommender system. In: *Proc. of the 15th Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, pp. 59-64 (2006)
52. Priester, J., Petty, R.: The gradual threshold model of ambivalence: relating the positive and negative bases of attitudes to subjective ambivalence. *Journal of Personality and Social Psychology* 71, 431-449 (1996)
53. Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., Riedl, J.: Grouplens: An open architecture for collaborative filtering of netnews. In: *Proc. of Computer Supported Cooperative Work*, pp. 175-186 (1994)
54. Resnick, P., Varian, H.R.: Recommender systems. *Communications of the ACM* 40, 56-58 (1997)
55. Richardson, M., Agrawal, R., Domingos, P.: Trust management for the semantic web. In: *Proc. of the Second International Semantic Web Conference*, pp. 351-368 (2003)
56. Schafer, J., Konstan, J., Riedl, J.: E-commerce recommendation applications. *Data Mining and Knowledge Discovery* 5, 115-153 (2001)
57. Sinha, R., Swearingen, K.: Comparing recommendations made by online systems and friends. *Proc. of the DELOS-NSF Workshop on Personalisation and Recommender Systems in Digital Libraries* (2001)
58. Swearingen, K., Sinha, R.: Beyond algorithms: an HCI perspective on recommender systems. *Proc. of SIGIR Workshop on Recommender Systems* (2001)
59. Tang, W., Ma, Y., Chen, Z.: Managing trust in peer-to-peer networks. *Journal of Digital Information Management* 3, 58-63 (2005)
60. Victor, P., De Cock, M., Cornelis, C., Pinheiro da Silva, P.: Towards a provenance-preserving trust model in agent networks. In: *Proc. of Models of Trust for the Web WWW2006 Workshop* (2006)
61. Victor, P., Cornelis, C., De Cock, M., Teredesai, A.M.: Key figure impact in trust-enhanced recommender systems. *AI Communications* 21, 127-143 (2008)
62. Victor, P., Cornelis, C., De Cock, M., Pinheiro da Silva, P.: Gradual trust and distrust in recommender systems. *Fuzzy Sets and Systems* 160, 1367-1382 (2009)
63. Victor, P., Cornelis, C., De Cock, M., Teredesai, A.M.: A comparative analysis of trust-enhanced recommenders for controversial items. In: *Proc. of the International AAI Conference on Weblogs and Social Media*, pp. 342-345 (2009)
64. Victor, P., Cornelis, C., De Cock, M., Teredesai, A.M.: Trust- and distrust-based recommendations for controversial reviews. *IEEE Intelligent Systems*, in press.
65. Zadeh, L.A.: Fuzzy sets. *Information and Control* 8, 338-353 (1965)
66. Zaihrayeu, I., Pinheiro da Silva, P., McGuinness, D.: IWTrust: Improving user trust in answers from the web. In: *Proc. of the Third International Conference On Trust Management*, pp. 384-392 (2005)
67. Zhang, S., Ouyang, Y., Ford, J., Makedon, F.: Analysis of a low-dimensional linear model under recommendation attacks. In: *Proc. of the International ACM SIGIR Conference*, pp. 517-524 (2006)
68. Ziegler, C., Lausen, G.: Propagation models for trust and distrust in social networks. *Information System Frontiers* 7, 337-358 (2005)
69. Ziegler, C., Golbeck, J.: Investigating correlations of trust and interest similarity - Do birds of a feather really flock together?. *Decision Support Systems* 43, 460-475 (2007)



## 组推荐系统

Judith Masthoff

**摘要** 本章将介绍如何通过聚合单个用户模型的信息和建立用户情感状态模型来完成面向用户组的推荐系统，并总结了该领域过去的研究成果。本章还介绍了群组推荐技术在单个用户推荐上的应用，特别是在解决冷启动(cold-start)问题和处理多重准则上的应用。

### 21.1 简介

到目前为止，推荐系统相关的大多数工作关注如何向单个用户推荐商品。例如，基于用户以前行为的偏好模型，为某个特定用户挑选一本书。推荐系统设计者通常面临的挑战是如何为单个用户决策出最佳物品。在这方面已经有很多的研究成果，书中其他章节(第 2~6 章)已经说明。

本章我们将进一步深入研究。在很多情况下，向群组用户推荐的效果可能优于向单个用户推荐的效果。例如，推荐系统可以根据所有群组成员的喜好进行建模，来为群组选择要看的电视节目或要听的一系列歌曲。面向群组的推荐甚至比面向单个的更加复杂，设想我们明确地知道单个用户的喜好，那么问题上升为如何整合所有单个用户的喜好来建立模型。在本章中我们将讨论群组推荐是如何工作的，面临哪些困难，以及已经取得了哪些成果。更加有趣的是，我们还将给大家展示群组推荐在单个用户推荐上的一些应用。因此，即使你正在开发的推荐系统是面向单个用户的，你仍然会乐于阅读本章(可能首先阅读 21.7 节会更加使你信服这一点)。

本章关注的重点是判断将何物推荐给群组用户，特别是如何整合单个用户模型。当建立一个群组推荐系统时，还有一些超出本章范围之外的事情需要考虑。特别是如下几点：

- **如何获得单个用户的偏好信息。**可以使用一些常用的推荐系统技术(如显式的评分，基于内容的协同过滤方法，见本书其他章节)。这里有一个困难就是，当群组用户使用这个系统时你很难推断出某个个体的喜好，但是在个体间一起使用联合概率模型可推断出个体喜好。另外一个困难就是单个用户的评分依赖于他所在的群组。例如，一个少年可能喜欢与他的弟弟妹妹一起观看某个节目，却不想和他的朋友一起观看。
- **系统如何知道当前使用对象是谁？**有不同的解决办法，例如，用户明确的登录、概率机制，利用时间预测当前对象，或利用符号和标签等[10]。
- **如何推出和解释群组推荐结果？**从相关章节可以看出，仅对单个用户进行推荐和解释推荐就已经存在很多困难，群组推荐的情况甚至更加困难，关于群组推荐的解释在文献[8]中和 21.8 节有进一步的讨论。

- **如何帮助用户做出最终的决定？** 在一些群组推荐实例中，提供给用户的是群组推荐，用户基于这些推荐结果来决定接下来该做什么，而在另外一些群组推荐中不存在这个问题（见 21.2.3 节介绍了积极和被动的群组的不同）。文献[8]中简要描述了哪些情况下可以为用户的决策提供帮助。

下一节将重点介绍群组推荐的使用场景，并根据不同的使用场景提供一种基于分类方法的群组推荐。21.3 节讨论了整合单个用户模型用于群组推荐的策略，并介绍哪些策略已经应用于当前系统和我们能够从过去的经验中学习得到什么。21.4 节解决为用户推荐一系列物品时的排序问题。21.5 节介绍情感状态模型，包括单个用户的情感状态如何被组内其他成员的情感状态所影响。21.6 节探讨如何通过情感模型来建立更加复杂的整合策略。21.7 节介绍了群组模型和群组推荐技术如何应用于单个推荐上。21.8 节对本章进行了总结并讨论了将来所面临的挑战。

## 21.2 应用场景和群组推荐系统分类

在很多环境下，我们需要适应一个群组的需求而不是单个用户的需求。下面，我们将提供两个场景为我们在该领域的工作提供灵感，讨论场景背后的相关工作，提供一种组推荐系统的分类方法，该方法的灵感来自使用场景的不同。

### 21.2.1 交互式电视

交互式电视使个性化的观赏体验成为可能，抛弃所有人都看同一个新闻节目的传统模式，取而代之的是为观众提供个性化的节目。例如，对于我来说，可能希望看更多的有关新西兰（我的家乡）、中国（多次在那里度假后令我着迷的国家）和足球的报道，而去掉关于板球（一个我无法理解的运动项目）和当地犯罪的报道。类似地，音乐节目可以选择一些我非常喜欢的音乐剪辑呈现给我。

在传统的推荐系统（如电脑软件）和交互式电视（如上面所简述的场景）之间存在两点主要的不同。首先，与使用个人电脑相比，观看电视的大多数时候是一个家庭或者社会团体。所以，电视新闻必须适应当前坐在它前面的一群人，而不是个别观众。第二，传统的推荐技术通常考虑的是向用户推荐一个特别的东西，例如，一部用户可能会观看的电影，而在上面所述的场景中，电视需要选择一系列的节目（如新闻特辑、音乐剪辑）给观众。将面向群组的推荐技术和推荐一系列物品结合是一件非常有趣的工作，因为为了使组中所有的单个用户都喜欢，对于推荐序列中某个特定用户不喜欢的物品，可能会允许你用序列中其他一些他们喜欢的物品来进行补偿。

### 21.2.2 环绕智能

环绕智能对于设定的屋里环境可以感知在场的人并做出反应。例如，假设这样一个书店，它里面的检测器可以检测有顾客在场并通过某种设备（例如，嵌入手机的蓝牙，装有射频标签的会员卡）确定顾客的身份。在该场景中，有各种传感器分布在书店的书架和各种零件中，这些传感器可以检测到在场的单个顾客。该书店可以将这些顾客的身份和他们的个人信息（如爱好、购买模式等）联系起来。

通过分布在场所中的这些设备，该书店可以提供给顾客一个变化迅速的环境，按顾客的效益最大化来增加书店销量。例如，广播在顾客群的听力范围内播放的背景音乐应该是充分

分析了顾客群的爱好后得到的。类似地,分布在书店的 LCD 显示器是基于附近的顾客来展示推荐的物品,商店展示窗口(展示新书目)的 LED 灯可以根据当时注视它的顾客的喜好和兴趣做出调整等。显然,在同一时间有多人出现的物理环境中经常需要群体推荐技术。

### 21.2.3 基于场景的推荐系统

本节将讨论最著名的组推荐系统:基于场景的推荐系统(scenarios underlying):

- MUSICFX[15]选择一个广播电台作为健身中心的背景音乐,使其适合在特定时间健身的人群,这与上面提到的环绕智能场景相似。
- POLYLENS[17]是 MOVIELENS 系统组推荐上的推广。MOVIELENS 基于单个用户的喜好来推荐影片,而单个用户的喜好是通过评分系统和群体过滤推断出的。POLYLENS 允许用户创建组并可以请求组推荐。
- INTRIGUE[2]通过考虑团体中成员(如孩子和残障人士等)的特点来向旅行团推荐旅游地点。
- TRAVEL DECISION FORUM[7]帮助一个团体实现一个共同所期望的度假计划。用户用一组特征集合(如运动和室内设施)标出他们的偏好,对于这些特征,系统合并单个用户的偏好,用户通过与代表组内其他成员的会话代理相互影响从而达到可接受的组偏好。
- 协同咨询旅游系统(CATS)[16]帮助用户选择度假方式。用户先考虑度假计划,并根据计划的特点做出评价(例如,“喜欢样图中所示的,但是缺个游泳池”)。基于这些评价,系统向他们推荐合适的度假计划。用户常常选择他们喜欢的度假物品使组内的其他成员看到,并标注出他们如何匹配组内每个成员的喜好(从他们的评价中得出)。可用单个成员的评价生成组偏爱模型,并基于这个模型给用户推荐其他的度假物品。
- YU 的电视推荐人[20]向群组推荐电视节目。它是基于单个用户喜欢的电视节目的特征(如类型风格、演员、关键词等)做出推荐的。

### 21.2.4 基于分类的群组推荐

和以上提出的场景在模式上有所不同,这是一种基于分类的组推荐技术:

- 单个用户的喜好是已知的,还是随着时间发生变化的。在多数场景中,组推荐从单个用户偏好着手,有时需考虑单个用户的喜好是随时间变化的,如在 CATS 系统中使用一种基于评论的方法,在第 13 章中讨论了这种评论和它在组推荐系统中扮演的角色。
- 推荐物品能被大家感受到,还是只作为选项展示。在交互式电视场景中,大家一起观看新闻。在环绕智能和 MUSICFX 场景中,人们一起欣赏音乐。相反地,在一些其他场景,他们提供给用户一个推荐列表。例如, POLYLENS 系统为组用户提供可能喜欢的电影的列表。
- 群组的特征是被动的,还是积极的。在大多数场景中,用户群并不需要像聚合单个用户偏好那样互动。但是,在 TRAVEL DECISION FORUM 和 CATS 中,用户群组通过共同参与构造组模型。
- 推荐单个物品,还是推荐一系列。在 MUSICFX、POLYLENS 和 Y U'S TV REC-

OMMENDER 的场景中，只推荐一个节目通常就足够了，因为人们通常每晚只看一部电影，电台可以一直不停播放，YU' S TV RECOMMENDER 也仅选择一部电视节目。类似地，TRAVEL DECISION FORUM 和 CATS 的用户一次只选择一个度假物品。相反地，在交互式电视使用场景中，用户被推荐一系列的电视节目，如编辑一个完整的新闻播报。同样，在 INTRIGUE 的应用场景中，一个旅行团在他们的旅途中会参观多个景点，所以他们可能会对一系列的景点感兴趣。与之类似的，在外界环境场景中用户可能会听多首歌曲或者在店内展示屏上查看多个条目。

在本章中，我们将关注包含以下特点的例子，即单个用户喜好已知，组用户直接体验的物品，群组特征是被动的，按系列推荐。系列推荐引出一些有趣的问题，即关注订阅序列(见 21.4 节)和考虑单个用户情绪状态(见 21.5 节和 21.6 节)。对于有鲜明主题经验的组来说，这点尤其重要，也就是，组内推荐这种方法是可行的。

DeCampos 等的基于分类的组推荐方法亦清晰地区分了被动组和积极组[4]。他还使用了另外两种特征：

- 单个用户的喜好是如何获得的。他们区分了基于内容和协同过滤两种情况。对于上面提到的诸多系统，POLYLENS 是唯一一个使用用户协同过滤这种方法的。
- 是对推荐结果合并还是对推荐模型合并。对于前者，先为单个用户产生推荐列表，然后再将这些列表合并入组推荐。对于后者，根据单个用户喜好将推荐模型合并生成组模型，然后用该组模型产生组推荐的结果。他们指出 INTRIGUE 和 POLYLENS 是对推荐结果合并，而其他推荐系统是对模型的合并。

这两个特征与组推荐系统如何执行相关，而不是使用场景所固有的。在本章中，我们将关注模型的合并，但是所提到的合并都指的是相同的合并策略。本章中列举的资料与单个用户喜好获得的方法无关。

21.3 合并策略

群组推荐需要解决的主要问题是当基于单个用户的喜好信息时，如何得出群组的评分。例如，假设这个群组包含 3 个成员：Peter、Jane 和 Mary。假设一个系统知道这三个人一起出现并知道他们各自的兴趣信息，这些信息分别包含在他们各自对物品评分列表中(如对音乐专辑或者广告的评分)。表 21.1 给出了单个用户对物品进行 10 分制评分的例子(1 代表非常讨厌，10 代表非常喜欢)，系统如何在指定的时间给出第四项条目(即群组评分)。

表 21.1 单个用户对 10 个物品评级的例子(A to J)

	A	B	C	D	E	F	G	H	I	J
Peter	10	4	3	6	10	9	6	8	10	8
Jane	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6

21.3.1 合并策略概览

现有很多将单个评分合并入群组评分的策略(例如，在一个政党领袖选举中的应用)。例如，“最小损失策略”采用评分中的最小值来避免为群组成员带来损失(见表 21.2)。受“社会选择”理论的启发，表 21.3 中列举了 11 种合并策略(具体细节见文献[10])。

表 21.2 最小损失策略示例

	A	B	C	D	E	F	G	H	I	J
Peter	10	4	3	6	10	9	6	8	10	8
Jane	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group Rating	1	4	2	6	7	8	5	6	3	6

### 21.3.2 合并策略在相关工作中的应用

大多数相关工作使用表 21.3 所列举的合并策略中的一种(通常伴随一些小的改进), 且在实际使用时会有所不同:

- INTRIGUE 使用一种加权模式的平均策略。该系统做出的群体推荐考虑了不同类型子群的偏好, 如儿童、残障人士等。该策略取平均值时, 其中的各权值基于各子群的人数和子群的相关性而定(儿童和残障人士被分配较高的相关性)。
- POLYLENS 使用最小损失策略, 它是假定一群人一起去看电影和最不快乐的成员相比总有一小群人是快乐的。
- MUSICFX 使用一个变种的平均无风险策略。用户对所有的广播站进行评分, 评分范围为+2(非常喜欢)到-2(非常讨厌)。将这些评分转换为正数(通过加 2 得到)并对它们求平方以扩大最受欢迎和最不受欢迎站点之间的分值差距。平均无风险策略用来产生一组列表: 只对评分高于某个阈值的物品求平均。为了避免饥饿现象(列表得不到更新)和总是选取同一个站点, 对列表中排在首位的站点进行随机加权选择。
- YU'S TV RECOMMENDER 使用一种变种的平均策略。它的群体推荐是建立在单个用户对节目特征的评分上的: -1 表示不喜欢的特征, +1 表示喜欢的特征, 0 表示中立。使团体的特征向量与单个成员的特征向量之间的距离最小, 这种方法类似于对每个特征的评分取平均。
- TRAVEL DECISION FORUM 运用了多种策略, 包括平均策略和中位数策略。中位数策略(不包含在表 21.3 中)使用评分的中间值。因此, 对于我们所举的例子, A 的群体评分结果为 10, F 为 9。中位数策略常因其不易被人为操纵而被选用: 使用者无法通过给出一个极端的评分(不能真实地反映他们的观点)将结果导向对自己有利的方向。相反, 如使用最小风险策略时, 不诚实的使用者可以通过给出一个极端负面的评价来避免得到他们只是稍微不喜欢的物品。操纵评价结果的问题在以下情况下显得非常重要: 当用户提供明确的评价时, 只针对群体进行推荐并且其他人的评价是可获得的, TRAVEL DECISION FORUM 就是其中的一个例子。在以下情况下会显得不那么重要: 评价是由用户行为推断得来的, 同时应用于单个用户推荐并且用户无法得知其他人的评价(即使使用了合并策略)。
- 在 CATS 中, 用户通过评论来标示一个度假物品应该包含哪些特征。对于确定的特征, 用户标示他们是否需要(如滑冰的需求)。对于另外的特征, 他们标示数量(如至少 3 个滑雪电梯)。群组模型包含所有用户的需求, 那些满足大多数需求的物品将获得推荐。用户也可以完全舍弃一些度假物品, 因此该策略包含无风险的一面。



表 21.3 合并策略方法综述

合并策略	如何运行	实例
简单多数投票	采用“得票最多者获胜”的原则：相类似，投票最多的物品被挑选	当 A 是群组里大多数人评分最高时，就选择 A，E 紧随其后（如果排除 A 后，E 的评分就是最高的）
平均	对所有单个用户的评分取平均值	B 的群组评分是 6，即 $(4+9+5)/3$
乘法	对所有单个用户的评分取乘积	B 的群组评分是 180，即 $4 \times 9 \times 5$
波达计数	根据物品在单个用户偏好列表中的排名计数，排名最后的物品得 0 分，靠前一位的得 1 分，以此类推	A 的群组评分是 17，即 $0$ (Jane 的最后一位) + $9$ (Mary 的第一位) + $8$ (Peter 的并列第 3 位)
科普兰规则	计算物品根据简单多数方法胜过其他物品的次数减去输给其他物品的次数	F 的群组评分是 5，因为 F 胜过了 7 个物品 (C, D, G, H, I, J)，输给了 2 个物品 (A, E)
通过投票	计算评分超过某个通过阈值（比如 6）的单个用户个数	B 的群组评分是 1，F 是 3
最小损失	选取单个用户评分的最小值	B 的群组评分是 4，即 4, 9, 5 中的最小值
最大值	选取单个用户评分的最大值	B 的群组评分是 9，即 4, 9, 5 中的最大值
无损平均	排除单个用户评分在某个阈值（如 4）以下的物品之后，取平均值	J 的群组评分是 7.3 (8, 8, 6 的平均值)，其中 A 被排除了，因为 Jane 讨厌它
公平	假设单个用户轮流选择物品来排名	物品 E 有可能被选为第 1 位 (Peter 打分最高的)，或者排在 F (Jane 打分最高的) 和 A (Mary 打分最高的) 之后
最尊者	用最受尊敬用户的评分	如果 Jane 是最受尊敬的人，则 A 的群组评分是 1。如果 Mary 是最受尊敬的人，则是 10

应该注意的是，YU’S TV RECOMMENDER 和 TRAVEL DECISION FORUM 在聚合每个特征的偏好时没有采用公平的想法：在某个特征上失分不会让你在另一个特征上得到补偿。

虽然 MUSICFX、POLYLENS 和 CATS 三者在某些评估方面有了进展，但对于这些系统中他们的策略是多么有效和如何根据不同的效果来选择不同的策略还没有得到研究，在 21.4 节的实验中我们给出了一些新的解释。

与此相反，Y U’S TV RECOMMENDER 已经给出了一些评估方法[20]。他们发现当组成员相对同质时，得到的聚合效果很好，但当组成员相对异质时结果却差强人意，这样的情况正如我们所期望的。如果组内成员非常相似，给出的平均策略将会使个人相当满意，但如果组内成员的喜好差别很大，那么结果将变得很差。

21.3.3 哪种策略效果最好

我们进行了一系列的实验来研究表 21.3 所列的策略中哪一种是最好的（细节详见文献 [10]）。

在实验 1 中（见图 21.1），我们研究人们以用户作为导向来评估是如何解决这类问题的 [13]。给参与者类似于表 21.1 中所列的单个用户的评分样本，这些评分是精心挑选的，以便在不同的策略间进行区分。参与者被问及这个群组应该看哪些节目，如果时间允许可以观看的节目数分别为 1, 2, …, 7。在合并策略下我们比较参与者的决策和理由，发现参与者更关注公平原则，防止损失和“饥饿”（“这一个是献给 Mary 的，因为到目前为止她还没有得到一件她喜欢的”）。参与者的行为反映出几种策略（如平均策略、最小损失策略

和无损平均策略), 而另外一些策略(如波达计数法、科普兰规则)却没有用到。



图 21.1 实验 1: 如果考虑系统任务, 人们会挑什么样的物品顺序呢

在实验 2(见图 21.2)中, 提供给参与者一个节目序列和表 21.1 中的单个用户评分, 其中节目序列是使用合并策略选出的。他们被要求评估该群组对于这个序列的喜欢程度, 并解释为什么这样评估。通过实验我们发现乘法策略(将单个用户评分相乘)的执行效果最佳, 在某种意义上参与者普遍认为它是唯一一种选出的节目序列可以使团体中所有成员都满意的策略, 波达计数、平均、无损平均和最大值这些策略也表现得比较好。还有一些策略(如科普兰规则、简单多数投票、最小损失)似乎应该被丢弃, 因为这些策略相对某些群组成员执行效果很差。



图 21.2 实验 2: 人们喜欢什么

我们也将参与者的评价与简单满足模型函数(simple satisfaction modelling functions)进行了比较。除了这些, 我们发现通过使用下面的方法可以得到更加准确的预测方法:

- 二次评价, 例如, 使评分为 9 分和 10 分之间的差距比 5 分和 6 分之间的大。
- 归一化, 考虑了人们会通过不同的角度进行评价, 例如, 有人使用比较极端的分值评价, 而有一些人使用比较中等的分值评价。

## 21.4 序列顺序的影响

在 21.2 节所提到, 我们对推荐物品的顺序很感兴趣。例如, 对于电视的个性化新闻, 推荐系统需要挑选出 7 个新闻节目播放给某个群组。

选择节目时, 可以使用聚合策略(如乘法策略)来组合单个用户的偏好值, 然后选出七组评分最高的新闻节目。当这些物品选好后, 剩下的问题是以什么样的顺序来播放这些节目。例如, 这些新闻的播放可以是按组评分的降序, 先是评分最高的物品, 最后是评分最低的。或者, 也可以把物品按随机顺序播放。

然而, 我们面临的问题在实际情况中更加复杂。首先, 在变化非常迅速的环境中, 组

成员身份不断变化,所以,以当前成员的身份决定接下来的七个物品似乎不是一个合理的策略,在最坏的情况下,第七个物品播放的时候,那些成员可能都已经离开电视了。

其次,对新闻节目播放列表的整体满意度更多取决于节目的播放顺序而不是其中人们想看的节目。例如,以获得最佳的满意度,我们可能需要确保新闻节目包含以下几个方面:

- **一个好的故事流。**把主题相关的节目一起播放可能是最好的。例如,如果我们有两则关于迈克尔·杰克逊的新闻(如他的葬礼和相关的颂词),把它们一起播放是最好的。同理,把所有的体育新闻凑在一块播放也是合适的。
- **情绪的一致性。**把相同情绪的新闻放在一起可能是最好的。例如,观众可能不希望看到在两个开心的新闻(如失业率下降和比赛胜利)中间出现一个悲伤的新闻(如一个士兵的死亡)。
- **令人深刻的结尾。**以一个大众喜欢的新闻结尾可能是最好的,因为这样观众最能记住这个新闻列表。

在其他推荐领域中也出现类似的推荐顺序问题。例如,一个音乐节目排序歌曲时要考虑它们的节奏。推荐系统可能需要额外的信息(如歌曲包含的心情、主题、节奏)来优化排序。讨论如何解决这些问题已经超出了本章的主题(而且这些跟推荐系统特定领域是密切相关的)。我们只是想强调,已经显示的物品可能影响在用户心中最好的下一个物品是什么。例如,假设音乐推荐系统推荐的前四首都是蓝调歌曲,那么排在第六位的蓝调可能比排在第五位的古典戏曲更适合排在下一位。

在实验3中(见图21.3),我们研究了前一条新闻可能会影响用户对下一条新闻的感受。其中包括,我们发现,情绪(从上一个物品产生的)和相关性会影响对后续物品的评分。这意味着,把个人信息聚合成群组信息是需要不断重复进行的,每次都要决定下一个要播放的物品是什么。

[这里插入你喜欢的运动俱乐部]赢得了一场重要比赛。  
Jennifer Lopez的豪华轿车车队100米的旅行计划。  
心脏病发生概率能减少一半吗?  
欧洲之神是否存在?  
地震袭击保加利亚。  
英国大火持续。  
保加利亚-西班牙足球赛况,随后是保加利亚3名主要球员受伤。

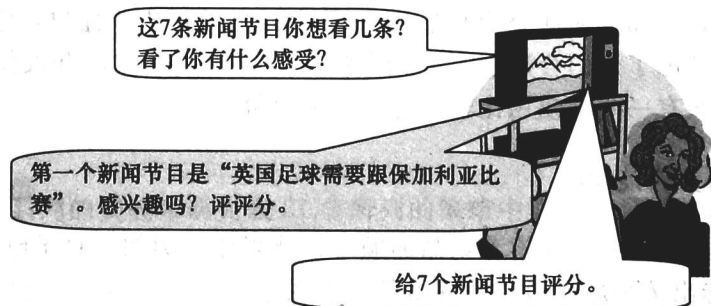


图 21.3 实验3: 研究新闻主题对心情的影响

## 21.5 对情感状态建模

为一组人做推荐的时候,你总不可能任何时候都给每个人喜欢的东西。但是,你也不

想有人对推荐结果非常不满意。例如,在一家商店中,如果一个客户要离开且不再光顾,是因为他们实在受不了背景音乐,这是一种不好的情形。许多商店目前选择播放一些没有人真正讨厌,但也不会是大多数人喜爱的音乐,这可能会避免失去客户,但不会导致销售增长。一个理想的商店会把音乐调整到顾客的听赏水平所能接受的范围区间内,在这个区间内挑选的歌曲都是顾客大多数时候喜爱的(有助于增加销售和顾客重复购买的可能性)。为了做到这点,不可避免客户有时也会听到他们不喜欢的歌曲,但发生的这一时刻顾客还是能忍受的(例如,因为上一首歌他们非常喜爱的,所以他还是处于很好的心情)。因此,持续地监控组成员的满意度是很重要的。当然,如果要顾客必须对他们接触到的音乐、广告等进行评分,这无疑会给顾客带来不能接受的负担。类似地,通过传感器(如心跳速率监视器或面部表情识别器)测量这种满足感还不是很可行的做法,因为这些做法对顾客太过滋扰,结果也不准确,而且成本昂贵。因此,我们建议对组成员的满意度建模,根据我了解到他们的喜好进行预测。

### 21.5.1 对个人的满意度进行建模

在文献[12]中,我们研究对个人满意度建模的四个满意度函数。我们比较了这些函数对于满意度的预测与实际用户的满意度。我们还完成了一项(见图 21.4)关于预测与用户真实感受实验的比较。

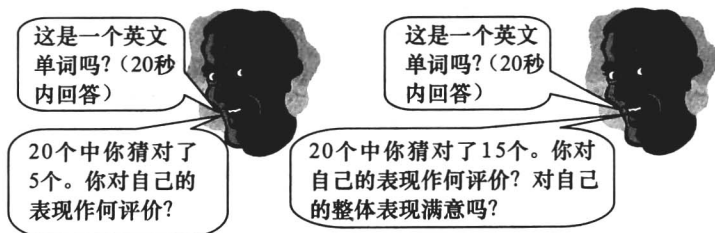


图 21.4 实验 4: 通过回答一系列问题得出整体满意度

满意度函数很好地定义了用户在看完一组物品序列后对一个新物品的满意程度,如下:

$$\text{Sat}(\text{items} + \langle i \rangle) = \frac{\delta \times \text{Sat}(\text{items}) + \text{Impact}(i, \delta \times \text{Sat}(\text{items}))}{1 + \delta} \quad (21.1)$$

其中,给定已有满意度  $s$  下,对新物品  $i$  满意度的影响定义为

$$\text{Impact}(i, s) = \text{Impact}(i) + (s - \text{Impact}(i)) \times \epsilon, \text{ for } 0 \leq \epsilon \leq 1 \text{ and } 0 \leq \delta \leq 1 \quad (21.2)$$

参数  $\delta$  代表了满意度随时间的衰减( $\delta=0$  代表过去的物品没有任何影响了,而  $\delta=1$  说明这个物品没有任何衰减)。

参数  $\epsilon$  代表了用户经历了前一个物品后,对新物品满意度的变化。这个参数是从心理学和经济学的文献中启发得来的,说明了心情确实会影响价值判断[12]。比如,半数参与电视问卷调查的人,如果先收到一个小礼物,他们的心情都很好。推荐给这些参与者的电视节目都有不错的效果。因此,如果一个用户由于喜欢之前的物品带来了好心情,下次出现了一个他不太喜欢的物品时,不好影响也通常较小(多少取决于  $\epsilon$ )。

参数  $\delta$  和  $\epsilon$  是依赖于用户的(在文献[12]的实验中证实了)。我们不会在本章中定义  $\text{Impact}(i)$ , 相关详细信息参考文献[12], 但它涉及二次评分和规范化如上面所讨论的实

验中发现的那样。

21.5.2 个人满意度对群组的影响

上述满意度函数并未考虑群组内其余成员的满意度，事实上他们的满意度也会影响个人的满意度。正如在社会心理学[12]中讨论过的，这里会发生两个主要的现象。

**情绪感染。**首先，其他用户的满意度可能会导致所谓的情绪感染：其他用户感到满意可能会提升其中某位用户的满意度（例如，如果有人向你微笑，你自然也会报以微笑，感觉会更好）。相反的情况也会发生：其他用户感到不满意可能也会降低某位用户的满意度。例如，如果你和你的朋友去看电影，当你的朋友明显不太喜欢这部电影的时候，这种情况可能会让你对该电影有一个负面的影响。

情绪感染可能取决于你的性格（有些人更容易被别人影响），以及与其他人的关系。人类学家和社会心理学家已经发现有四种基本类型的社交关系，见图 21.5。在实验 5（见图 21.6）中，我们证实了情绪感染确实取决于你的关系：你更容易被某些你爱的人感染（如你最好的朋友）或你尊重的人感染（你的母亲或老板），这种感染程度超过平等的或者与之竞争的关系。



图 21.5 人际关系的四种基本类型

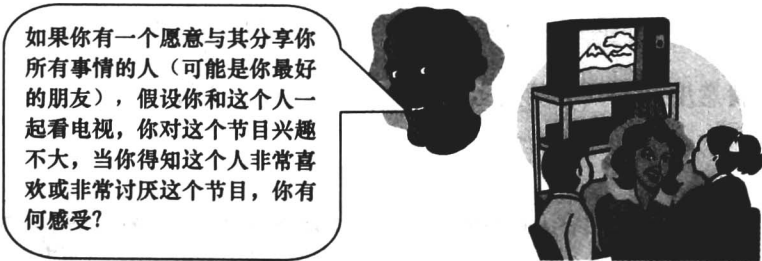


图 21.6 实验 5：情绪感染对关系类型的影响

**从众心理。**其次，其他用户的观点可能会影响你自己表达的观点，这是基于所谓从众心理。

图 21.7 显示了著名的由 Asch[3]开展的从众心理实验。与会者提出了一个很容易的事，如判断四条线段中的哪一条与卡片 A 中的线段方向相同：参与实验者认为他们被其他参与者所包围，但实际上其他人是实验团队成员，而这些成员会事先约定选择相同的错误



答案。结果表明，大多数参与实验者也会选择同样的错误答案。



图 21.7 从众心理实验(来自 Asch)

有两种从众类型：1)规范化影响，即你希望成为集团的一分子，所以赞同组内其他成员的意见，即使跟你内心的信念并不一致；2)信息影响，你的观点改变是因为你认同集体观点是对的。信息影响会改变你的满意度，而规范化影响可以通过情绪感染改变其他人的满意度，因为你描述了一种不诚恳的情绪。

更多复杂的满意度函数可参见文献[12]，相关研究对情绪感染和两种从众心理类型进行了建模。

## 21.6 情感状态在合并策略中的使用

一旦建立了准确的个人满意度模型，最好使用该模型改善组合并策略。例如，这种合并策略可以满足群组中满意度最低的成员。这可以由许多不同方式实现，而我们才刚刚开始研究该问题，例如：

- **强支持最挑剔用户策略。**该策略是挑选满意度最低的用户最喜欢的物品。如果存在多个这样的物品，则使用标准合并策略，如乘法策略，来进行区分。
- **弱支持最挑剔用户策略。**该策略是挑选满意度最低的用户比较喜欢的物品，比如，评分在 8 分或者以上的物品。此时使用标准合并策略，如乘法策略，在这些物品中进行选择。
- **加权策略。**该策略是根据用户满意度来给用户分配权重，然后使用一个加权形式的标准合并策略。比如，表 21.4 显示当使用了平均策略时，分配 2 倍权重给 Jane 的效果。注意，权重不能应用于某些策略，如最小损失策略。

文献[14]中详细地讨论了这一点，提出一种基于代理的架构来将这些创意应用在智能场景，并描述了实现的原型。显然，在探索合并策略中使用情感状态的最佳方式时需要实证研究。

表 21.4 为 Jane 分配相同权重和两倍权重后的平均策略结果

	A	B	C	D	E	F	G	H	I	J
Peter	10	4	3	6	10	9	6	8	10	8
Jane	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Average(equal weights)	7	6	4.3	7.3	8.7	8.7	5.7	7.7	6.7	7.3
Average(Jane twich)	5.5	6.8	5.3	8.3	8.3	8.8	5.8	8	5.8	7.5

21.7 对单个用户进行组推荐

如果你打算开发一个为单个用户进行组推荐的应用又该如何做？组推荐技术在三个方面产生效用：1)合并多准则；2)解决所谓的冷启动问题；3)考虑其他的观点。第 22 章也探讨了在给个人推荐时聚合的必要性，并包括了几种具体的聚合函数。

21.7.1 多准则

由于问题是多维度的，有时候很难给出推荐：此时多重标准发挥作用。例如，在新闻的推荐系统中，用户可能有地理位置的偏好(对离家近的报道更感兴趣，或者与他们热衷的度假地区相关的报道)。用户可能更喜欢最近的新闻，或者有主题偏好(比如，偏向于政治新闻而不是体育新闻)。推荐系统可能得到类似于表 21.5 的结果，这里不同的新闻在不同的标准下评分不同。那么该推荐哪个新闻报道呢？

表 21.5 基于不同标准下对 10 个新闻报道的评分

	A	B	C	D	E	F	G	H	I	J
Topic	10	4	3	6	10	9	6	8	10	8
Location	1	9	8	9	7	9	6	9	3	8
Recency	10	5	2	7	9	8	5	6	7	6

多准则的问题将在第 24 章中详细讨论。这里将讨论在组推荐系统中如何处理这些问题。

除了有多个标准而非多个用户要满足，表 21.5 与上述的组推荐类似(表 21.1)。对于该问题，我们可以应用组推荐技术。然而，适应一群人和适应一组标准之间有着重要区别。为了适应一群人，公平地对待每一个人似乎是合理的和道德上正确的。当然，可能有一些例外，例如，当组中包含成人和儿童，或者当它是某人的生日时。但总的来说，平等似乎是一个不错的选择，而且也被用于上述的组适应策略。与此相反，适应一组标准时，这里没有特别理由来假定所有标准都是重要的。甚至可能对一个特定的人而言，并不是所有的标准都同样重要。事实上，实验发现用户对待标准的方式不同，会对某些标准给予更多的重视(例如，最近行为比位置更重要)[11]。那么，该如何使用组推荐策略来处理这种问题呢？有多种方法可以做到：

- 把策略应用于最重视的标准上。不重要的评分标准可以完全忽略。例如，假定“位置”标准被认为是不重要的，那么它的评分就可以忽略。表 21.6 显示了忽略位置时的平均策略结果。
- 把策略应用于所有的标准且使用权重。在不重要的标准上的评分可以赋予较小的

权重。例如，在平均策略上，某个标准的权重乘以该标准的评分，从而产生新的评分。举例来说，假设主题和近期效应标准的重要性三倍于位置标准的重要性。表 21.7 显示了使用了这些权重的平均策略结果。若使用乘法策略，评分与权重相乘没有任何影响。在乘法策略中，把权重作为指数因子更好，因此把权重作为评分的幂，得到新的值取代原有评分。请注意，在上述两种策略中，零权重会导致评分被完全忽略。

- **调整策略使重要与非重要的标准表现不同：**无损不等均值。重要标准的损失可以避免，但非重要标准则不能。假设标准位置再次被认为是不重要的。表 21.8 显示了阈值为 6 的无损不等均值策略的结果。

表 21.6 剔除了不重要的位置标准的平均策略

	A	B	C	D	E	F	G	H	I	J
Topic	10	4	3	6	10	9	6	8	10	8
Recency	10	5	2	7	9	8	5	6	7	6
Group	20	9	5	13	19	17	11	14	17	14

表 21.7 主题和近期效应分配权重为 3 及位置权重为 1 时的平均策略

	A	B	C	D	E	F	G	H	I	J
Topic	30	12	9	18	30	27	18	24	30	24
Location	1	9	8	9	7	9	6	9	3	8
Recency	30	15	6	21	27	24	15	18	21	18
Group	61	36	23	48	64	60	39	51	54	50

表 21.8 阈值为 6 且位置标准不重要的无损不等均值策略

	A	B	C	D	E	F	G	H	I	J
Topic	10	4	3	6	10	9	6	8	10	8
Location	1	9	8	9	7	9	6	9	3	8
Recency	10	5	2	7	9	8	5	6	7	6
Group	21			22	26	26		23	20	22

有证据表明，人们的行为反映了这些策略的结果[11]。然而，在该领域要知道哪些策略是最好的，还需要更多的研究。此外，还需要更多的研究来确定一个标准是“不重要”的。

多准则问题也是本手册其他章节(见第 24 章)讨论的主题。

### 21.7.2 冷启动问题

推荐系统的一大问题是所谓的冷启动问题：为适应用户，系统需要知道用户过去喜欢什么。这在基于内容过滤推荐中是必要的，用来选定与用户喜欢的物品相似的物品。在社会化推荐中，决定和当前用户相似的用户在过去时间里对相同物品的喜恶是必要的(见图 21.8)。如果你对用户一无所知，因为用户才刚刚开始使用推荐系统，该怎么办？推荐系统的设计者倾向于用以下两种方法解决问题：要么让用户一开始就对物品进行评分，要么让他们回答一些人口统计学方面的问题(然后再使用固有的模式作为起点，如老人喜欢古典音乐)。

这两种方法都需要用户花费精力。挑选哪些物品让用户去评分也不是容易的事情,某些固有模式有可能是错误的并具有冒犯性的(有的老人喜欢流行音乐,而且并不喜欢被归类为老年人)。

本章所提到的组推荐研究提供了另一种解决方案。对于推荐系统新用户,为其提供使得现有群体中用户都满意的推荐。假设该用户与某个已存在的用户相似,虽然不知道是哪一个用户,既然推荐结果能让群体中的用户都快乐,新用户同样也会快乐的。

我们将逐步地了解该新用户的偏好,例如,通过分析他们对推荐物品的评分,或者更隐含一点,通过分析他们在推荐结果上花的时间或花不花时间。我们提供使得已有用户和新用户都开心的推荐结果给该新用户(或者更确切地说,被假设为新用户的人)。刚开始赋予新用户的权重较小,因为对他们了解不多,然后再逐渐增加权重。我们也开始分配较小的权重给一些现有用户,这些现有用户的品味现在显著不同于该新用户。

图 21.9 展示了一个适应的例子:该系统包括观察到的新用户在一定程度上品味,并开始减少一些其他用户的权重。长期使用该系统后,用户的推断愿望将彻底主宰物品选择。我们使用了 MovieLens 的数据集对该方法的有效性进行了小规模的研究。

我们随机选了 5 部电影和 12 位评价过这些电影的用户:其中推荐系统已知的有 10 人,新用户为 2 人。在已知用户组使用乘法策略,得到新用户的电影排名。结果令人鼓舞:在电影中排名最高的电影,其实是新用户最喜欢的电影,同时根据新用户的信息,其他电影的排名也还可以。使用权重可进一步改进排名方法,同时权重开始反映新用户和已知用户的相似性。文献[9]中给出了研究和应用群体适应策略来解决冷启动问题的更多细节。

### 21.7.3 虚拟组成员

最后,组适应也可以用于个人,做法是把虚拟的人物加入组里。例如,对于小孩子看电视娱乐,父母并不反对,但也希望小孩子能偶尔学到点什么。当小孩一个人的时候,父母信息可以作为虚拟组成员加入小孩的组里面,这样电视节目可以同时满足父母和孩子的需求。

## 21.8 总结与挑战

组推荐是一个相对较新的研究领域。本章的目的是介绍该领域,尤其是聚合个人用户信息方面。欲了解更多信息请参见文献[10, 12, 14, 11, 9, 7, 8]。



图 21.8 在社会化过滤中的冷启动问题



图 21.9 逐步学习这位用户及她与哪位用户最相似

21.8.1 提出的主要问题

本章提出的主要问题有：

- 适应群体在许多情况下是必要的，如互动电视、智能环境、推荐旅游团等。在场景间差异的激励下，可以使用多个维度对组推荐进行分类。
- 许多策略是为了聚合个人偏好(见表 21.3)，其中有的策略表现比其他策略要好一些。用户似乎关心避免损失和公平性。
- 现有的组推荐与分类维度中和聚合策略中的组推荐不同。见表 21.9 的概述。

表 21.9 组推荐系统

系统	应用场景	分类				应用策略
		已知偏好	鲜明主题经验	组激活	推荐序列	
MUSICFX[15]	根据锻炼的项目为健身中心健身的人选择电台	是	是	否	否	无损均值
POLYLENS[17]	为一组提供电影观看	是	否	否	否	最小损失
INTRIGUE[2]	为一基于子组(例如)特征的组用户提供旅游景点参观	是	否	否	是	均值
TRAVEL DECISION FORUM[7]	提供一个共同的假期计划所需属性的组模型并争取一组用户的同意	是	否	是	否	中位数
YU's TV RECOMMENDER[20]	基于多属性的个人评分为一组用户提供一个电视节目观看	是	否	否	否	均值
CATS[16]	基于推荐节日的个人评价帮助用户选择一个共同的节日	否	否	是	否	满足重要需求 无损使用
MASTHOFF's GROUP RECOMMENDER[10, 12]	根据个人评分为一组用户选择唱片序列	是	是	否	是	乘法策略

- 当推荐一系列物品时，在序列的每个步骤中都要聚合个人信息，因为前面的物品可能会影响后面物品的评分。
- 构建满意度函数来预测单个用户在一个序列中的任意时间的满意度是可行的。然而，组互动效果(如情绪感染和整合)会使该过程变得复杂。
- 在实验中评估聚合策略和满意度函数表现如何是可行的，虽然这并不是一个简单的问题。
- 组聚合策略不仅在为一组人提供推荐时是重要的，也可以用于个人推荐，比如，在避免冷启动问题和处理多重标准时。

21.8.2 警告：组建模

术语“组建模”也可以用于与本章所展示的完全不同的工作中。组成员之间的常识建模(如文献[6, 19]，对组间互动建模(如文献[18, 5])和基于个人模型的群组构成(如文献[18, 1])，这些方面已经开展了很多研究。



### 21.8.3 面临的挑战

相比个人推荐的工作来说，组推荐仍是一个全新的领域。本章介绍的工作只是一个起点。未来的研究还有许多具有挑战性的方向，其中包括：

- **推荐物品序列给一个群组。**我们的工作似乎是目前唯一着手通过均衡序列来解决公平问题的。即使序列对于复杂的使用场景是重要的，其工作还是没有研究序列的平衡，也没有研究序列的顺序。显然，在推荐和顺序序列方面还需要更多的研究，尤其是在已展示的物品应如何影响其他物品方面。其中一些研究必须在推荐系统的特定领域进行。
- **情感状态的建模。**这里仍需要大量的工作去获得经过检验的满意度函数。本章和文献[12]中提出的工作只是一个起点。特别是大规模评估及群组规模的影响力研究是必需的。
- **聚合策略中整合情感状态。**正如 21.6 节所示，有许多方法把情感状态用于一个聚合策略中。我们提出了这方面的一些初步想法，但大量的实证研究需要进一步深入。
- **组推荐的解释：透明度和隐私。**有人可能认为准确预测个人的满意度能改进推荐系统的透明度：显示其他组成员的满意度能提高用户对推荐过程的理解，或许还可以使得用户更容易地接受他们不喜欢的物品。然而，用户的隐私需求很可能与透明度需求冲突。组推荐系统的一项重要任务是避免尴尬。用户一般喜欢与组成员保持一致以避免被不喜欢（正如 21.5.2 节所讨论的一样，其他组成员的情感状态如何影响个人的情感状态）。在文献[12]中，我们研究不同的组聚合策略会如何影响隐私。对组推荐的解释还需要更多的工作，特别是关于如何平衡透明度、隐私和可理解度。第 15 章提供了推荐系统中不同角色解释的详细内容。
- **用户界面设计。**良好的用户界面设计会增加个人对组推荐的满意度。例如，在显示一个物品时，可以给用户显示下一个物品（如在电视节目通过字幕告知）。这可能会告知用户谁对下一个节目兴趣度比当前节目更大一点。
- **冷启动问题的组聚合策略。**在 21.7.2 节，我们简单介绍了利用组聚合策略去解决冷启动问题。然而，这方面的研究非常少，而且需要大量的工作去验证和优化这种方法。
- **处理不确定性。**在本章中，假设我们获得了准确的个人偏好档案。例如，在表 21.1 中，推荐系统明确 Peter 对于 B 物品的评分为 4。然而在现实中，我们通常面对的是概率数据。例如，我们认为 Peter 的评分有 80% 的概率为 4。在采用聚合策略的时候需要考虑到这一点。DeCampos 等试图通过使用贝叶斯网络去处理这种不确定性[4]。然而，他们迄今专注于平均和复数投票策略的研究，尚未解决避免损失和公平性问题。
- **实证研究。**为了推动这一领域的前进，更多的实证评估是至关重要的。在真实环境下设计一套容易控制且能大规模进行的实证评估是我们面临的挑战，特别是在处理组推荐和情感状态时。至今所有的研究（包括我自己的），仅局限在小范围内，且都是人为的设定或缺乏控制。

### 致谢

Judith Masthoff 的部分研究得到 Nuffield 基金会的支持，No. NAL/00258/G。

## 参考文献

1. Alfonseca, E., Carro, R.M., Martn, E., Ortigosa, A., Paredes, P.: The Impact of Learning Styles on Student Grouping for Collaborative Learning: A Case Study. *UMUAI* 16 (2006) 377-401
2. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Tailoring the Recommendation of Tourist Information to Heterogeneous User Groups. In S. Reich, M. Tzagarakis, P. De Bra (eds.), *Hypermedia: Openness, Structural Awareness, and Adaptivity*, International Workshops OHS-7, SC-3, and AH-3. *Lecture Notes in Computer Science* 2266, Springer Verlag, Berlin (2002) 280-295
3. Asch, S.E.: Studies of independence and conformity: a minority of one against a unanimous majority. *Psychol. Monogr.* 70 (1956) 1-70
4. de Campos, L.M., Fernandez-Luna, J.M., Huete, J.F., Rueda-Morales, M.A.: Managing uncertainty in group recommending processes. *UMUAI* 19 (2009) 207-242
5. Harrer, A., McLaren, B.M., Walker, E., Bollen L., Sewall, J.: Creating Cognitive Tutors for Collaborative Learning: Steps Toward Realization. *UMUAI* 16 (2006) 175-209
6. Introne, J., Alterman, R.: Using Shared Representations to Improve Coordination and Intent Inference. *UMUAI* 16 (2006) 249-280
7. Jameson, A.: More than the Sum of its Members: Challenges for Group Recommender Systems. *International Working Conference on Advanced Visual Interfaces*, Gallipoli, Italy (2004)
8. Jameson, A., Smyth, B.: Recommendation to groups. In: Brusilovsky, P., Kobsa, A., Njedi, W. (Eds). *The Adaptive Web Methods and Strategies of Web Personalization*. Springer (2007) 596-627
9. Masthoff, J.: Modeling the multiple people that are me. In: P. Brusilovsky, A. Corbett, and F. de Rosi (eds.) *Proceedings of the 2003 User Modeling Conference*, Johnstown, PA. Springer Verlag, Berlin (2003) 258-262
10. Masthoff, J.: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers. *UMUAI* 14 (2004) 37-85
11. Masthoff, J.: Selecting News to Suit a Group of Criteria: An Exploration. 4th Workshop on Personalization in Future TV - Methods, Technologies, Applications for Personalized TV, Eindhoven, the Netherlands (2004)
12. Masthoff, J., Gatt, A.: In Pursuit of Satisfaction and the Prevention of Embarrassment: Affective state in Group Recommender Systems. *UMUAI* 16 (2006) 281-319
13. Masthoff, J.: The user as wizard: A method for early involvement in the design and evaluation of adaptive systems. *Fifth Workshop on User-Centred Design and Evaluation of Adaptive Systems* (2006).
14. Masthoff, J., Vasconcelos, W.W., Aitken, C., Correa da Silva, F.S.: Agent-Based Group Modelling for Ambient Intelligence. *AISB symposium on Affective Smart Environments*, Newcastle, UK (2007)
15. McCarthy, J., Anagnost, T.: MusicFX: An Arbiter of Group Preferences for Computer Supported Collaborative Workouts. *CSCW*, Seattle, WA. (1998) 363-372
16. McCarthy, K., McGinty, L., Smyth, B., Salamo, M.: The needs of the many: A case-based group recommender system. *European Conference on Case-Based Reasoning*, Springer (2006) 196-210
17. O' Conner, M., Cosley, D., Konstan, J.A., Riedl, J.: PolyLens: A Recommender System for Groups of Users. *ECSCW*, Bonn, Germany (2001) 199-218. As accessed on <http://www.cs.umn.edu/Research/GroupLens/poly-camera-final.pdf>
18. Read, T., Barros, B., Brcena, E., Pancorbo, J.: Coalescing Individual and Collaborative Learning to Model User Linguistic Competences. *UMUAI* 16 (2006) 349-376
19. Suebnukarn, S., Haddawy, P.: Modeling Individual and Collaborative Problem-Solving in Medical Problem-Based Learning. *UMUAI* 16 (2006) 211-248
20. Yu, Z., Zhou, X., Hao, Y., Gu, J.: TV Program Recommendation for Multiple Viewers Based on User Profile Merging. *UMUAI* 16 (2006) 63-82



# 高级算法

- 第 22 章 推荐系统中的偏好聚合
- 第 23 章 推荐系统中的主动学习
- 第 24 章 多准则推荐系统
- 第 25 章 具有健壮性的协同推荐

## 推荐系统中的偏好聚合

Gleb Beliakov、Tomaso Calvo 和 Simon James

**摘要** 本章对聚合函数在推荐系统中的应用进行了概述。类似于算术平均值等简单聚合函数在聚合用户特征、物品评分和相似性指标等方面得到了广泛应用，然而，其他一些聚合函数在许多系统中能提供更高的精度和灵活度。本章描述了聚合函数在推荐领域的一些重要成员和属性的定义，以及成熟的创建方法和很多例子。

### 22.1 简介

偏好、标准和相似性的聚合贯穿着推荐系统的许多阶段。通常都用算术平均或最大/最小函数完成聚合，而许多能够为更多相关推荐提供灵活性和适应性的聚合函数常常被忽视。本章概述了聚合函数的基本知识和它们的属性，介绍了一些重要的成员，其中包括平均值(generalized mean)、Choquet 与 Sugeno 积分(Choquet and Sugeno integral)、有序加权平均(ordered weighted averaging)、三角形准则(triangular norms and conorm)以及双极聚合函数(bipolar aggregation function)。类似的方法可以为输入、连接、取消连接和混合行为之间的各种交互进行建模。接下来，我们将介绍基于分析公式、算法或者实验数据的不同聚合函数组合方式。讨论在保留重要属性的同时，如何使聚合函数的参数符合观测数据。通过使用更复杂适合的函数代替算术平均值，消除输入的冗余，可以改善自动推荐的质量，并使推荐系统适合特定的领域。

### 22.2 推荐系统中的聚合类型

总体来说，推荐系统(RS)引导用户在经过电子化的物体和信息的巨大数据库中找到自己喜欢的物品。推荐系统利用大量的用户特定方案生成个性化推荐，其个性化物品子集展示的目标在概念上区分了与推荐系统类似的方法，如互联网过滤。从出现开始，推荐系统的使用迅速分布在现有的应用中，例如，推荐电影[34]、网页[5]、新闻[36]、医疗[14, 31]、音乐和其他产品[32, 40]。

很明显，用来推荐物品的方案依赖特定的应用，以及数据被收集和在系统中被使用的方法。基于与物品特性相关的方案的推荐一般分类为基于内容的推荐(CB)，而利用用户相似度的推荐称为协同过滤(CF)[1, 2]。进一步将基于人口统计的过滤方法(DF)、基于效用的(UB)和基于知识的(KB)[16]推荐方法与通常使用物—物相似度的内容推荐方法区别开是有益的。最近的文献的特点是专注于混合推荐系统(HS)，它结合了两种或两种以

---

Gleb Beliakov and Simon James, School of Information Technology, Deakin University 221 Burwood Hwy, Burwood 3125, Australia, e-mail: gleb@deakin.edu.au, sjames@deakin.edu.au

Tomaso Calvo, Departamento de Ciencias de la Computación, Universidad de Alcalá 28871-Alcalá de Henares(Madrid), Spain. e-mail: tomaso.calvo@uah.es

翻译：北京大学-王雪丽，马占国 审核：韶关学院-梁永霖老师，吉林大学-丁彬钊



上的方法。

**协同过滤方法**用物品偏好或者相似用户的评分作为推荐的理由。这种类型的推荐一直被成功地应用在像亚马逊[32]一样的电子商务网站上,其中通过相似的品味能更好地推断兴趣,而不是通过模糊的主观的物品描述。假设一个非主流类别,如独立音乐,因其不同于主流而被轻易定义。由于流派覆盖范围广泛的风格,独立音乐的艺术之间可能相似性很小,除了他们的粉丝<sup>①</sup>。聚合函数(一般为简单或者加权平均)经常被用来聚合评分或者相似用户的偏好,然而它们还可以用来确定用户相似度,帮助定义邻居(可以参照本书的第4章)。

**基于内容的过滤方法**通过匹配物品属性与用户信息来构造方案。例如,一个新闻推荐系统可能会为每个拥有关键词的用户构建用户信息,对未知物品的兴趣可以通过来自与用户信息中相对应的故事的关键词来预测。基于内容的推荐中聚合函数的使用方法基于每个用户特定的用户信息特性和物品描述。这里考虑它们在物品评分计算、相似性计算和用户信息构建中的使用。

**人口统计过滤技术**根据用户信息为每个用户分配一个人口统计分类。每个人口统计分类有一个相关的用户原型或者用户典型用来为推荐构建方案。与物品历史不同,用户相似度更可能是由用户个人信息计算得来的,所以维度比大多数协同过滤技术低。这使得最近邻或者其他分类聚类方法尤其管用。

**基于效用的推荐系统**并不构建长期模型而是匹配物品与用户当前的需求,并考虑他们的普遍倾向和偏好。例如,一个用户正在寻找一本特定的书,从用户的过去行为中已知其偏爱旧精装版,即使需要花很长时间才能买到。在基于内容过滤的情况下,系统中描述物品可以根据其特性,更具体的是与特性相关的效用。由于是基于内容的过滤,即使用户信息和系统信息都可能不同,仍然可以应用聚合函数。

**基于知识的推荐系统**使用相关和类似物品的背景知识推断用户的需求以及如何最大化满足他们的需求。基于知识的方法并不仅利用典型的相似性方法,如相关性,还利用了用户可能感兴趣的属性相似性。例如,当用户表明他喜欢电影 A Good Year,基于知识的推荐系统可能会将其与电影 A Beautiful Mind(也是由 Russell Crowe 主演的)或者 Jeux d'Enfants(也是由 Marion Cotillard 主演的)相关联。由于历史表明用户偏好法国电影,系统会推断该用户喜欢由 Marion Cotillard 主演的电影 A Good Year,并做出相应的推荐。[16]中指出基于知识的推荐系统经常借鉴基于用例的推理方法。

**混合推荐系统**用来克服每个推荐方法自身的缺陷,Burke[16]将其分类为加权型、合并型、转换型、特征组合型、级联型、特征递增型、元层次型。聚合函数也可以用在混合过程中。例如,组合加权型混合推荐中不同的推荐分数或特征组合型混合推荐中的特征。另外,其中一些混合推荐方法对于在不同阶段使用的聚合函数执行的改善有很大帮助。例如,瀑布型方法使用其中一个过滤技术来减少数据集的大小,而特征递增型混合推荐可能使用其中一种方法来降维。协同过滤推荐使用的相似性方法可以使用在基于效用和内容的推荐框架中构建的用户特定聚合函数(如权重与参数之间的相似性)。相似的元层次型混合推荐在文献[16]中有描述。转换型混合推荐的转换标准在一定程度上可能会用到聚合函数,然而,合并型混合推荐就很少用到了。

① 有趣的是,独立音乐的粉丝因缺乏流行文化与消费主义的一致性,容易成为使用协同过滤推荐的电子商务网站的市场目标。最近的一篇文献[26]中有描述。

聚合函数需要多个输入并将它们合并为一个代表性的输出。关于聚合函数的简单例子,包括算术平均值、中位数、最大最小值。在推荐系统中使用更复杂和更具表达性的方法通常的动机是更准确地推荐,然而,在某些情况下与其他数据处理方法相比,聚合函数是比较实用的选择。在接下来的小节中我们将调查聚合函数在不同类型的推荐系统中扮演的角色,这些角色可以显示它们可以被用在哪里以及已经应用在了哪里。

### 22.2.1 协同过滤中的偏好聚合

给定一个用户  $u$  和相似用户的邻居  $U_k = \{u_1, u_2, \dots, u_k\}$ , 用户  $u$  对未知物品的偏好可以通过聚合给定的分数进行预测。我们用  $R(u, d_i)$  来表示兴趣、评分或偏好的预测程度:

$$R(u, d_i) = \sum_{j=1}^k \text{sim}(u, u_j) R(u_j, d_i) \quad (22.1)$$

该方程式称为加权算术平均(WAM), 其中用户与其相似用户的相似度  $\text{sim}(u, u_j) = w_j$  是权重,  $R(u_j, d_i) = x_j$  是要聚合的输入。给定  $w_j, x_j \geq 0$ ,  $R(u, d_i)$  是一个聚合函数。选择 WAM 容易解释, 可以满足许多有用的属性并且在计算成本低的同时, 其他的聚合函数包括幂均值(可以是非线性)或者 Choquet 积分(针对相关输入)可能会更精确地预测用户评分。

### 22.2.2 CB 与 UB 推荐中的特性聚合

配置文件可表示为特性偏好向量  $P_u = (p_1, \dots, p_n)$ , 物品的描述可以根据物品满足特性的程度, 如  $d_i = (x_1, \dots, x_n)$ 。此时,  $x_j = 1$  表示该物品完全满足偏好。也可以是一个关键字向量, 在此情况下  $x_j = 1$  仅表示关键词只出现一次。一个物品的整体得分  $R(u, d_i)$  便由  $x_j$  的聚合决定:

$$R(u, d_i) = f(x_1, \dots, x_n) \quad (22.2)$$

式(22.2)是一个聚合函数, 该函数提供满足特定边界条件的方法, 且相对于  $x_j$  的增加是单调的。 $R(u, d_i)$  分数可以用来排名后面要推荐的未知物品。如果推荐系统只允许展示一个物品, 那么如何评估该评分及其原因都变得很重要。如果用户只想购买或浏览满足他们所有偏好的物品, 那么应该使用像最小化一样的合取函数。另一方面, 如果其中一些偏好不能同时满足, 例如, 用户对戏剧和恐怖片感兴趣, 那么平均函数或析取函数可能比较可靠。我们将在 22.3 节举出很多这种聚合函数广泛分类的例子。

在计算物品与物品相似度可行的情况下, 基于内容的过滤液可以使用与协同过滤相对应的方法[2]。在这种情况下, 用户信息可能包含所有或一部分之前评分/购买的物品  $D = \{d_1, \dots, d_q\}$ , 相似性在未知物品  $d_i$  和  $D$  中的物品中进行计算:

$$R(u, d_i) = \sum_{j=1, (j \neq i)}^q \text{sim}(d_i, d_j) R(u, d_j) \quad (22.3)$$

在这种情况下, 基于内容的方法可以受益于使用聚合函数来确定物品相似度和在 22.2.4 节中提到的物品邻居。

### 22.2.3 CB 与 UB 的配置文件构建

更复杂的系统将为每个偏好分配一个权重。为了加强在线体验, 许多推荐系统趋向于

从在线行为中学习偏好(和权重),而不是要求用户显式地表达。之前评分或购买的物品的特性可以聚合成为每个偏好的整体得分。给定一个偏好  $p_j$ , 让  $x_{ij}$  为物品  $d_i$  满足  $p_j$  的程度, 那么分数  $w(p_j)$  可以表示为

$$w(p_j) = f(x_{1j}, \dots, x_{nj}) \quad (22.4)$$

一旦确定了所有的偏好,  $w(p_j)$  就可以用来确定  $w_j$ , 用于计算式(22.2)。

#### 22.2.4 物品和用户相似度以及邻居的形成

使用式(22.1)的推荐行为和准确率将很大程度上依赖相似度是如何确定的(加权向量)。用户之间的相似度可以通过之前评分或购买的物品来计算, 或者利用与用户相关的已知特性进行计算, 如年龄、位置和用户已知的兴趣。最常用的计算相似性的方法, 如式(22.1)中的加权, 使用了余弦计算[39]和皮尔逊相关系数[36]。最近, 其他相似性计算方法相继出现, 像模糊距离[4]和基于用户评分分布的(可以参照本书第4章)其他特定推荐指标[18, 3]。

如式(22.1)也可以考虑用在最近邻(kNN)方法的框架中。聚合函数已被用于提高最近邻规则的精确性和效率, 同时 OWA 和 Choquet 积分为框架提供模拟衰减权重和邻居交互的功能[45, 12]。在最近邻设置中, 相似性无异于多维近似或距离。欧几里得距离用于计算在文献[42]中使用评分和个人信息作为输入的推荐系统的相似度。欧几里得距离只是一种类型的度量, 可能不能很好地捕捉到距离的概念, 例如, 数据维度之间在某种程度上是相关的甚至是不可比较的。在某些聚合函数的帮助下定义的指标包括 OWA 算子和 Choquet 积分, 已经在文献[41, 13]中探讨, 并可能在计算某些推荐系统的相似性时有用。

如果将式(22.1)中的  $\text{sim}(u, u_j)$  看作权重而不是相似度, 那么我们可以认为各种聚合函数的权重确定问题已经被广泛研究。一个方法是通过最小二乘法拟合技术从数据子集中学习权重。例如, 给定一组交互评分物品  $D = \{d_1, \dots, d_q\}$ , WAM 的权重可以由下式来模拟:

$$\begin{aligned} & \text{minimize } \sum_{i=1}^q (R(u, d_i) - \sum_{j=1}^k w_j R(u_j, d_i))^2 \\ & \text{s. t. } \quad \quad \quad w_j \geq 0 \quad \quad \quad \forall j \\ & \quad \quad \quad \sum_{j=1}^k w_j = 1 \end{aligned}$$

真正被确定的是可最小化剩余误差的权重向量  $w = (w_1, \dots, w_k)$ 。那么其中每个权重就是在精确预测  $R(u, d_i)$  时给定用户的重要性。非线性函数, 如加权几何平均, 也可以用这种方法模拟。该算法因其计算时间效率相对较高, 且可以由离线计算得到或者在运行时根据推荐系统和数据集大小计算获得。

另外, 聚合函数可以用来组合不同的相似性计算方法。给定一些数量的相似性计算方法  $\text{sim}_1(u, u_1), \text{sim}_2(u, u_1), \dots, \text{sim}_n(u, u_1)$  等, 可以获得一个整体的相似性计算方法<sup>⊖</sup>。此类聚合相似度在文献[20]中用来推荐电影。在此例中, 余弦与相关性得分通过该方法进行结合, 该方法是非线性结合的聚合函数。

⊖ 针对基于多重标准的相似度, 参见第 24 章。

## 22.2.5 基于实例推理的连接词在推荐系统中的应用

在模糊集社区许多研究者的方法是基于用例推理[23]来构建推荐问题的，其中聚合函数可以作为连接词。这导致了如下的形式规则：

$$\text{If } d_{i1} \text{ is } A_1 \text{ AND } d_{i2} \text{ is } A_2 \text{ OR } \dots d_{in} \text{ is } A_n \text{ THEN } \dots \quad (22.5)$$

$x_1, x_2, \dots, x_n$  表示规则判断  $d_{i1}$  是  $A_1$  等的满足程度，而聚合函数用来代替 AND 和 OR 操作。例如，若一个用户的属性显示一个对喜剧与动作片的偏好，那么该用户的推荐规则可能包括“如果该电影是喜剧或者动作片，那么推荐它”<sup>①</sup>。每个风格都可以表示为一个具有模糊链接的模糊集，用来聚合满足度。OR 和 AND 行为通常分别由析取和合取聚合函数建模。在推荐系统中，可以看出巨大的加强的特性是可取的[44, 9]。若有需要，这个特性允许进行多种大幅有效调整生成一个健壮的推荐系统，或者许多微小的调整来减少推荐。

构建式(22.5)的方法可以用于 CB、UB 和 KB 中物品与配置文件或队列的匹配。在某些基于人口的推荐系统中，在一个给定的类中，物品基本上会推荐给每个人，这使得分类过程为该推荐系统的首要任务。根据用户满足若干原型的程度将用户分类是可取的，反过来可以根据用户对每个物品的兴趣来描述物品。例如，一个免息的个人贷款会非常吸引大学生，对新手妈妈也会有些吸引，但是对新娘燕尔可能就没什么吸引力了。如果系统聚合了每个人口信息的兴趣值，用户可能会部分满足每个原型。这导致了类似式(22.5)的规则。“IF 物品对于学生感兴趣 OR 对于新手妈妈感兴趣，那么用户  $u$  也将对其感兴趣”，或者“如果用  $u$  是未婚而且要么是学生要么是妈妈，那么为其推荐该物品”。

## 22.2.6 加权混合系统

给定一组通过不同方法得到的推荐分数，例如， $RCF(u, d_i)$ ， $RCB(u, d_i)$  等，整体得分可以通过使用聚合函数  $f$  得到：

$$R(u, d_i) = f(R_{CF}(u, d_i), R_{CB}(u, d_i), \dots) \quad (22.6)$$

P-Tango 系统[19]使用协同过滤与基于内容得分的线性组合进行推荐，而且根据推断出的用户偏好来调整权重。两个或多个方法的聚合可以通过使用多种不同属性和行为的方法来执行。使用非线性或者更加复杂的方法可以优化某些推荐系统的排名过程，产生非相关性更小、精确度更高的预测。

## 22.3 聚合函数概论

聚合函数的目的是组合输入，这些输入一般被表示为在模糊集中的关系度、偏好度、证据强度或假设的支持度等。本节将在介绍一些广为人知的成员之前先提供一些初步的定义。

### 22.3.1 定义和属性

我们要考虑的是定义在单元区间的聚合函数  $f: [0, 1]^n \rightarrow [0, 1]$ ，但是其选择也是可

① 我们在此也要注明这样的规则可以用于任何推荐系统来决定何时推荐物品，比如，“如果用户不活跃，则推荐某个东西”。

能的。输入为 0 表示没有关系、偏好、证据、满足等，自然  $n$  个 0 的聚合结果为 0。同样地，值 1 表示完全关系(最强程度的偏好和证据)，1 的聚合结果自然为 1。

聚合函数还需要在每个参数上单调，即输入的增加并不能降低整体分数。

**定义 22.1** (聚合函数) 聚合函数是  $n > 1$  个参数从  $(n$  维)单元立方到单元区间的映射  $f: [0, 1]^n \rightarrow [0, 1]$ ，并具有以下属性：

$$(i) \underbrace{f(0, 0, \dots, 0)}_{n\text{-times}} = 0 \text{ and } \underbrace{f(1, 1, \dots, 1)}_{n\text{-times}} = 1$$

$$(ii) x \leq y \text{ implies } f(x) \leq f(y) \text{ for all } x, y \in [0, 1]^n$$

对于某些应用，输入可能具有各种组成部分(例如，一些值可能会缺失)。特别是在自动系统情况下，可能希望使用在  $n=2, 3, \dots$  个基本属性相同的参数上定义的功能来提供一致的聚合结果。然后，符合下列定义[33]的功能才值得考虑。

**定义 22.2** (扩展聚合函数) 扩展聚合函数是如下一个映射

$$F: \bigcup_{n \in \{1, 2, \dots\}} [0, 1]^n \rightarrow [0, 1]$$

该映射到具有确定  $n$  值的域  $[0, 1]^n$  上的限制是一个  $n$  元聚合函数  $f$ ，其中  $n=1$  时  $F(x)=x$ 。

聚合函数分类取决于它们与输入之间关系的整体表现[17, 21, 22]。在某些情况下我们需要高输入来补偿低输入，或者可以平均相互输入。在其他情况下，这样做是有意义的，高分互相加强而低分基本上被丢弃。

**定义 22.3** (分类) 聚合函数  $f: [0, 1]^n \rightarrow [0, 1]$ ：

平均 如果边界为  $\min(x) \leq f(x) \leq \max(x)$ ；

合取 如果边界为  $f(x) \leq \min(x)$ ；

析取 如果边界为  $f(x) \geq \max(x)$ ；

混合 其他情况下。

聚合函数的分类取决于推荐系统的输入是如何表示的，以及输出需要的灵敏度或广泛性。当在 CF 中聚合推荐得分时，使用聚合函数保证了物品的兴趣预测代表分数的总体趋势。另一方面，某些混合聚合函数的语义让它们更可能被使用。例如，MYCIN[14]是一个典型的专家系统，用来诊断和治疗稀有的血液疾病，使用了一个混合聚合函数，因此高分输入互相加强，而得分低于某一阈值的输入被惩罚。

聚合函数可以满足多种研究过的属性，使得它们可以在某种情况下有用处。这里我们提供那些文献中经常引用的定义。

**定义 22.4** [属性] 聚合函数  $f: [0, 1]^n \rightarrow [0, 1]$  是：

幂等 如果对于每个  $t \in [0, 1]$  其输出为  $f(t, t, \dots, t) = t$ ；

对称 如果其值不依赖参数的排列，例如，对于  $f(x_1, x_2, \dots, x_n) = f(x_{P(1)}, x_{P(2)}, \dots, x_{P(n)})$  每个  $x$  及  $(1, 2, \dots, n)$  的每个排列  $P = (P(1), P(2), \dots, P(n))$ ；

关联 如果对于  $f: [0, 1]^2 \rightarrow [0, 1]$ ，在所有  $x_1, x_2, x_3$  上  $f(f(x_1, x_2), x_3) = f(x_1, f(x_2, x_3))$  不变；

平移不变 如果对于所有  $\lambda \in [-1, 1]$  和  $x = (x_1, \dots, x_n)$ ，只要  $(x_1 + \lambda), \dots, (x_n + \lambda) \in [0, 1]^n$  并且  $f(x) + \lambda \in [0, 1]$ ，则公式  $f(x_1 + \lambda, \dots, x_n + \lambda) = f(x) + \lambda$  成立；

齐次 如果对于所有  $\lambda \in [0, 1]$  和  $x = (x_1, \dots, x_n)$ ，公式  $f(\lambda x_1, \dots, \lambda x_n) = \lambda f(x)$  成立；



**严格单调** 如果  $x \leq y$  但  $x \neq y$ , 则  $f(x) < f(y)$ ;

**Lipschitz 连续** 如果存在整数  $M$ , 对于任意两个输入  $x, y \in [0, 1]^n$ , 公式  $|f(x) - f(y)| \leq M_d(x, y)$  成立, 其中  $d(x, y)$  是  $x, y$  之间的距离。 $M$  的最小值叫作  $f$  的 Lipschitz 常数;

**具有中立元素** 如果存在一个  $e \in [0, 1]$  使得对于任何位置的  $t \in [0, 1]$  有  $f(e, \dots, e, t, e, \dots, e) = t$ ;

**具有吸收元素** 如果存在一个  $a \in [0, 1]$  使得对于任意  $x$  且  $x_j = a$  有  $f(x_1, \dots, x_{j-1}, a, x_{j+1}, \dots, x_n) = a$ 。

### 22.3.1.1 RS 中的实用注意事项

阐述许多重要并且被广泛研究聚合函数的正式定义之前, 我们先通过一些例子来讨论其中每个属性的某些影响。

**幂等** 所有平均聚合函数包括 22.3.2 节所定义的均值、OWA 和 Choquet 积分都是幂等的<sup>⊖</sup>。

通常该属性的解释趋向于解释为输入之间的一致性。然而在某些 RS 应用中, 例如, 在 CF 聚合评分时, 物品相对排名比输入/输出解释的通约性更重要。

**例 22.1** 几何平均  $G(x, y) = \sqrt{xy}$  是幂等的, 而结果  $T_p(x < y) = xy$  不是幂等的。对于任意两个对象  $d_1 = (x_1, y_1)$  和  $d_2 = (x_2, y_2)$ , 若  $G(d_1) > G(d_2)$  则说明  $T_p(d_1) = T_p(d_2)$ 。

**例 22.2** 取  $d_1 = (0.5, 0.5)$ ,  $d_2 = (0.2, 0.8)$ 。使用几何中项以及结果来聚合给定的  $G(d_1) = 0.5$ ,  $G(d_2) = 0.4$ ,  $TP(d_1) = 0.25$ ,  $TP(d_2) = 0.16$ 。如果  $d_i$  是 CF 中的物品得分, 在使用  $G$  的情况下, 最好将输出解释为用户  $u$  的预测评分。如果  $d_i$  是 UB 过滤中通过满足两个用户偏好的满足度进行的物品描述, 那么能更好地表示整个用例, 因为我们希望满足大多数偏好。

**对称性** 对称性经常用来表示相关输入的同等重要性。加权或未加权的准算术平均值可根据情况使用。虽然有序加权平均函数(OWA)是根据加权向量定义的, 但输入经过了非递增地预排序, 因此它的对称性与  $w$  无关。

**例 22.3** 协同 RS 认为一个物品由三个相似的用户  $d_i = (0.2, 0.7, 0.5)$  进行评分。我们考虑使用 OWA 函数或加权算数平均(WAM)加权向量  $w = (0.6, 0.3, 0.1)$ 。假设使用 WAM, 权重表示用户  $u_1$  与用户  $u$  之间很相似, 进一步有  $\text{sim}(u, u_1) > \text{sim}(u, u_2) > \text{sim}(u, u_3)$ 。因为用户  $u_1$  不是特别喜欢该物品, 在这种情况下的聚合分数为  $R(u, d_i) = \text{WAM}(d_i) = 0.6(0.2) + 0.3(0.7) + 0.1(0.5) = 0.38$ 。如果使用 OWA, 权重的一个解释是如果有一个或者两个相似用户喜欢该物品那么用户  $u$  也会喜欢, 不管哪个相似用户喜欢。那么有  $R(u, d_i) = \text{OWA}(d_i) = 0.6(0.7) + 0.3(0.5) + 0.1(0.2) = 0.59$ 。

**关联性** 关联性对于自动计算有所帮助, 因为它允许函数在任何维度递归的定义。这对协同 RS 的稀疏性问题可能有用。同样的函数可能用来评估一个由 10 个相似用户评分的物品以及被 1000 个相似用户评分的另一个物品。三角模型和三角余模型、统一模型和零模型都是关联的, 但是准算术平均值方法不是关联的。

⊖ 因为单调性的要求, 幂等性和求均的做法对于聚合函数是等价的。这个属性有时可以看作一致性, 因为当输入全体一致时, 输出和每个输入相同。

**例 22.4** 一个协同过滤 RS 使用个人信息来确定两个用户之间的相似性(例如,不需要在每次一个新物品被评分时重新估计值)。系统并没有为每个用户存储一个物品与用户的矩阵,而是使用一个统一模型  $U(x, y)$  来聚合相似用户的评分,存储一个聚合的物品评分  $\mathbf{d}=(U(d_1), \dots, U(d_n))$  的单一向量。当添加了一个新物品评分  $x_{ij}$ , 系统聚合  $U(U(d_i), x_{ij})$  并代替  $U(d_i)$  进行保存。好处是在更新预测评分时,不需要知道之前的得分及物品被评分的次数。

**平移不变性和均匀性** 平移不变性和均匀性方法的主要好处是转化或扩大研究领域并不会影响聚合输入的相对顺序。加权算术平均值、OWA、Choquet 积分都是平移不变的,因此只要输入是可比较的,不管输入是在文献[0, 100]还是文献[1, 7]上都没有影响。

**严格单调性** 严格单调性用在给可用来显示给用户的物品有限的应用中。当  $w_j > 0$ ,  $\forall_j$  时,加权算术平均和 OWA 函数是严格单调的,而算术平均和调和平均方法在  $\mathbf{x}$  上是严格的。不严格的聚合函数,如最大化方法,不能区分物品  $d_1=(0.3, 0.8)$  和物品  $d_2=(0.8, 0.8)$ 。

**例 22.5** 一个度假推荐网站使用基于效用的 RS, 其中 Łukasiewicz 三角余模型  $SL(x, y)=\min(x+y, 1)$  用来聚合物品特性。通过邮件通知用户每个  $SL(d_i)=1$  的物品是可以的。 $d_1=(0.3, 0.8)$  还是  $d_2=(0.8, 0.8)$  是没关系的,因为二者的预测完全满足用户的需求。

**Lipschitz 连续** 一般而言,连续性保证了输入的小误差不会导致输出的急剧变化。该特性在输入不管是物品描述还是用户评分不精确的推荐系统中尤其重要。一些函数只违反这个属性领域的一小部分(参照例 22.6)。只要在考虑推荐得分时考虑该特性,该函数可能依然适合。

**例 22.6** 算术平均方法  $G(x, y)=\sqrt{xy}$  不如 Lipschitz 特性,因为当一个输入接近 0 时,变化率是没有边界的。另外,调和平均数方法  $H(x, y)=\frac{2xy}{x+y}$  (在二元实例中)是 Lipschitz 连续的,其中 Lipschitz 常数  $M=2$ 。

**中立性和吸附性元素** 吸附性元素在 RS 中可保证某些物品经常或永不被推荐。例如,一个 UB 推荐系统可以移除每一个特性得分为 0 的物品,或者明确地推荐完全符合用户某个偏好的物品。三角模型或三角余模型都具有吸附性元素。将具有中立性元素的函数组合到聚合用户评分的推荐系统中(CF 或者 CB 架构中)允许指定不会影响推荐分数的值。例如,许多人都喜欢的一部电影其得到称赞的整体评分会因一个不怎么喜欢但依然被要求评分的用户的评分而降低。如果存在一个中立值,该评分就不会影响聚合分数。

## 22.3.2 聚合成员

### 22.3.2.1 准算术平均值

加权准算术平均方法成员包括幂平均值,这反过来包括了其他典型的均值方法,如算术和几何平均的特殊用例(可以参照文献[15]中的方法概述)。

**定义 22.5** (加权准算术平均) 给定一个严格单调且在  $g: [0, 1] \rightarrow [-\infty, +\infty]$  上连续被称为生成函数或生成器的函数以及一个权重向量  $\mathbf{w}=(w_1, \dots, w_n)$ , 加权准算术平均值为

$$M_{\mathbf{w},g}(\mathbf{x}) = g^{-1}\left(\sum_{i=1}^n w_i g(x_i)\right) \quad (22.7)$$

其中  $\sum w_j = 1$  且任给  $j$  有  $w_j \geq 0 \forall j$ 。

特殊用例包括:

算术平均  $\text{WAM}_w = \sum_{j=1}^n w_j x_j, g(t) = t$ ;

几何平均  $G_w = \prod_{j=1}^n x_j^{w_j}, g(t) = \log(t)$ ;

调和平均  $H_w = \left( \sum_{j=1}^n \frac{w_j}{x_j} \right)^{-1}, g(t) = \frac{1}{t}$ ;

幂均值  $M_{w,[r]} = \left( \sum_{j=1}^n w_j x_j^r \right)^{\frac{1}{r}}, g(t) = t^r$ ;

术语平均值通常用来表示平均行为。准算术平均值依据一个对所有  $w_j = 1/n$  要么对称要么不对称的权重向量来定义。通常给一个特定的输入分配一个权重表明了这个特定输入的重要性。所有幂平均方法(包括  $\text{WAM}_w$ 、 $G_w$  和  $H_w$ )在闭区间  $[0, 1]^n$  是幂等的、齐次的、严格单调的,但是只有加权算术平均是平移不变的。几何平均是非 Lipschitz 连续的 5。

### 22.3.2.2 OWA 函数

有序加权平均函数(OWA)也是平均聚合函数,它并没有将权重与一个特定的输入结合,而是跟与其他输入的相对值或顺序相结合。Yager[43]对此有介绍而且在模糊集社区非常流行。

**定义 22.6** (OWA) 给定一个权重向量  $w$ , OWA 函数如下:

$$\text{OWA}_w(x) = \sum_{j=1}^n w_j x_{(j)}$$

其中  $(\cdot)$  符号代表组成部分  $x$  以非递增顺序  $x(1) \geq x(2) \geq \dots \geq x(n)$  排列。

依据权重向量  $w$ , OWA 操作的特殊分类包括:

**算术平均** 所有的权重相等,例如,  $w_j = 1/n$ ;

**最大化函数** 其中  $w = (1, 0, \dots, 0)$ ;

**最小化函数** 其中  $w = (0, \dots, 0, 1)$ ;

**中位数函数** 其中  $w_j = 0$ , 所有  $j \neq m$ , 如果  $n = 2m + 1$  是奇数, 则  $w_m = 1$ , 其中  $w_j = 0$ , 所有  $j \neq m, m + 1$ 。如果  $n = 2m$  是偶数, 则  $w_m = w_{m+1} = 0.5$ ;

OWA 函数是一个分段线性幂等聚合函数。如果对于任意  $j$  有  $w_j > 0$ , 则 OWA 是对称的、齐次的、平移不变的、Lipschitz 连续的和严格单调的。

### 22.3.2.3 Choquet 与 Sugeno 积分

Choquet 和 Sugeno 积分被称为模糊积分是依据一个模糊度量定义的平均聚合函数。其在构建输入值  $x_j$  间的交互中 useful。

**定义 22.7** (模糊度量) 给定  $\mathcal{N} = \{1, 2, \dots, n\}$ 。一个离散的模糊度量是在  $v: 2^{\mathcal{N}} \rightarrow [0, 1]$  上单调(如当  $A \subseteq B$  时  $v(A) \leq v(B)$ )的, 而且满足  $v(\emptyset) = 0$  和  $v(\mathcal{N}) = 1$  的集合方法。给定任意两个集合  $A, B \subseteq \mathcal{N}$ , 模糊度量可以是:

**加法** 对于  $v(A \cap B) = \phi$ , 有  $v(A \cup B) = v(A) + v(B)$ ;

**对称** 其中  $|A| = |B| \rightarrow v(A) = v(B)$ ;

**次模** 如果  $v(A \cup B) - v(A \cap B) \leq v(A) + v(B)$ ;

**超模** 如果  $v(A \cup B) - v(A \cap B) \geq v(A) + v(B)$ ;

**次加性** 如果  $A \cap B = \emptyset$  时, 有  $v(A \cup B) \leq v(A) + v(B)$ ;

**超加** 如果  $A \cap B = \emptyset$  时, 有  $v(A \cup B) \geq v(A) + v(B)$ ;

**可分解** 给定函数  $f: [0, 1]^2 \rightarrow [0, 1]$ , 如果  $A \cap B = \emptyset$  时有  $v(A \cup B) = f(v(A), v(B))$

**Sugeno( $\lambda$ -模糊度量)** 如果  $v$  可分解, 对于  $\lambda \in [-1, \infty]$ , 有  $f = v(A) + v(B) + \lambda v(A)v(B)$

Sugeno 与 Choquet 积分的表现取决于关联模糊度量的值和属性。用于定义 Choquet 积分的模糊度量可以被解释为权重分配, 不是仅分配给单个输入而是分配给输入子集。输入之间可能有冗余或者某些输入之间相辅相成。

**定义 22.8** (Choquet 积分) 给定一个模糊度量  $v$ , 离散的 Choquet 积分由下式给出:

$$C_v(x) = \sum_{j=1}^n x_{(j)} [v(\{k | x_k \geq x_{(j)}\}) - v(\{k | x_k \geq x_{(j+1)}\})] \quad (22.8)$$

其中  $(.)$  符号代表组成部分  $x$  以非递减顺序  $(x(1) \leq x(2) \leq \dots \leq x(n))$  排列 (注意这与 OWA 相反)。

Choquet 积分的特殊分类包括加权算术平均和模糊度量分别为加法或对称的 OWA 函数。次模(submodular)模糊度量导致凹的 Choquet 积分, 结果是增加低输入对函数的影响大于增加高输入的影响。相反地, 超模(supermodular)模糊度量导致凸的函数。当  $A(B \rightarrow v(A) < v(B))$  时, Choquet 积分是幂等的、齐次的、平移不变的、严格单调的。如果模糊度量是对称的, 则函数将明显地满足对称属性。

Choquet 积分曾在数值输入中占主要优势, 接下来要定义的 Sugeno 积分对计数输入很有用, 而它的定义也用到了模糊度量。

**定义 22.9** (Sugeno 积分) 给定一个模糊度量  $v$ , Sugeno 积分可表示为

$$S_v(x) = \max_{j=1, \dots, n} \min\{x_{(j)}, v(H_j)\} \quad (22.9)$$

其中  $(.)$  表示一个非递减的输入排列, 如  $(x(1) \leq x(2) \leq \dots \leq x(n))$  (与 Choquet 积分相同) 和  $H_j = \{(j), \dots, (n)\}$ 。

为更好地理解 Choquet 和 Sugeno 的表现已经引进了一些指标。特别是 Shapley 值给出了一个给定输入整体重要性的指标, 同时两个输入间的交互指标显示了它们冗余和相辅相成的程度。

**定义 22.10** (Shapley 值) 给定  $v$  为模糊度量。对于每个  $i \in N$  的 Shapley 指标为

$$\varphi(i) = \sum_{A \subseteq N \setminus \{i\}} \frac{(n - |A| - 1)! |A|!}{n!} [v(A \cup \{i\}) - v(A)]$$

Shapley 值为向量  $\varphi(v) = (\varphi(1), \dots, \varphi(n))$ 。

**定义 22.11** (交互指标) 给定  $v$  为模糊度量。对于每对  $i, j \in N$  交互指标如下:

$$I_{ij} = \sum_{A \subseteq N \setminus \{i, j\}} \frac{(n - |A| - 2)! |A|!}{(n - 1)!} [v(A \cup \{i, j\}) - v(A \cup \{i\}) - v(A \cup \{j\}) + v(A)]$$

若交互指标为负, 两个输入之间有冗余。当值为正时, 输入会以某种程度互相加强而且它们现在的权重的价值超过它们各自的权重结合起来的值。

#### 22.3.2.4 三角模型与三角余模型

合取和析取聚合函数的典型例子分别是所谓的三角模型与三角余模型(t-norms and t-

conorms)[28]。给定任何三角模型  $T: [0, 1]^2 \rightarrow [0, 1]$ , 三角余模  $S$  可以用一个二元函数表示:

$$S(x, y) = 1 - T(1 - x, 1 - y)$$

反之亦然。因此, 三角模型与三角余模型经常被并行研究, 因为许多  $S$  相关的属性可以由  $T$  来确定。中立元素  $e=1$  时三角模型是关联的、对称的, 而中立元素  $e=0$  时三角余模型是关联的、对称的。下面是四个基本的三角模型与三角余模型的定义。

**定义 22.12** (四个基础的三角模型) 四个基本的三角模型的二元用例如下:

最小化  $T_{\min}(x, y) = \min(x, y)$ ;

结果  $TP(x, y) = xy$ ;

**Lukasiewicz 三角模型**  $TL(x, y) = \max(x + y - 1, 0)$ ;

激烈积  $T_D(x, y) = \begin{cases} 0, & (x, y) \in [0, 1]^2 \\ \min(x, y), & \text{其他} \end{cases}$ 。

**定义 22.13** (四个基本的三角余模型) 四个基本的三角余模型的二元用例如下:

最大化  $S_{\max}(x, y) = \max(x, y)$ ;

概率和  $SP(x, y) = x + y - xy$ ;

**Lukasiewicz 三角模型**  $SL(x, y) = \min(x + y, 1)$ ;

激烈积  $S_D(x, y) = \begin{cases} 1, & (x, y) \in ]0, 1]^2 \\ \max(x, y), & \text{其他} \end{cases}$ 。

有参数的三角模型与三角余模型的成员包括上面的特殊和有限制的用例。这些成员是基于生成函数来定义的, 称为阿基米德三角模型。

**定义 22.14** (阿基米德三角模型) 如果对于每个  $(a, b) \in [0, 1]^2$  都有  $n = \{1, 2, \dots\}$  使得  $T(a, \dots, a) < b$ , 那么三角模型及为阿基米德三角模型。

对于三角余模型不等式是相反的, 例如, 三角余模型  $S > b$ 。连续的阿基米德三角模型可以使用它们的生成器来表示

$$T(x_1, \dots, x_n) = g^{(-1)}(g(x_1) + \dots + g(x_n))$$

其中  $g(1)=0$  则  $g: [0, 1] \rightarrow [0, \infty]$  是连续的、严格递减的函数, 是  $g$  的反矩阵, 如

$$g^{(-1)}(x) = g^{-1}(\min(g(1), \max(g(0), x)))$$

阿基米德成员包括 Schweizer-Sklar、Hamacher、Frank、Yager、Dombi、Aczel-Alsina、Mayor-Torrens 和 Weber-Sugeno 三角模型和三角余模型。

### 22.3.2.5 零模型和统一模型

在某些情况下, 也许需要高输入值互相加强, 而低输入值拉低整体输出。换句话说, 聚合函数对于较高值是析取的, 对于较低值是合取的, 还有可能取平均值, 如果一些值比较高、一些值比较低。这是较高值被解释为“正”信息, 低值被解释为“负”信息的典型用例。

在其他情况下, 有可能要聚合高低值使得输出趋向于一个中间值。因此此类聚合函数需要在它们不同的阶段为合取的、析取的或平均的。而统一模型和零模型就是这种聚合函数的典型例子, 但也有许多其他的模型。本章提供以下定义。

**定义 22.15** (零模型) 零模型是一个二元聚合函数  $V: [0, 1]^2 \rightarrow [0, 1]$ , 是关联的、对称的, 因此存在一个元素  $a$  属于开区间  $[0, 1]$  使得



$$\forall t \in [0, a], V(t, 0) = t$$

$$\forall t \in [a, 1], V(t, 1) = t$$

**定义 22.16** (统一模型) 统一模型是二元聚合函数  $U: [0, 1]^2 \rightarrow [0, 1]$ , 是关联的、对称的并且拥有一个中立元素  $e$  属于开区间  $[0, 1]$ 。

一些统一模型可以像准算术平均和阿基米德三角模型类似的方法, 使用生成函数来构建。这些为可解释的统一模型。

**定义 22.17** (可解释的统一模型) 给定  $u: [0, 1] \rightarrow [-\infty, +\infty]$  是一个严格递增的双射验证  $g(0) = -\infty$ 、 $g(1) = +\infty$  以致对于某些  $e \in [0, 1]$ , 有  $g(e) = 0$ 。

$$\bullet U(x, y) = \begin{cases} g^{-1}(g(x) + g(y)), & (x, y) \in [0, 1]^2 \setminus \{(0, 1), (1, 0)\} \\ 0, & \text{其他} \end{cases}$$

是一个具有中立元素  $e$  的合取统一模型, 称为可解释的合取统一模型。

$$\bullet U(x, y) = \begin{cases} g^{-1}(g(x) + g(y)), & (x, y) \in [0, 1]^2 \setminus \{(0, 1), (1, 0)\} \\ 1, & \text{其他} \end{cases}$$

是一个具有中立元素  $e$  的析取统一模型, 称为可解释的析取统一模型。

$3-\pi$  函数是一个可解释统一模型的实例[46]。它使用了生成函数  $g(x) = \ln\left(\frac{x}{1-x}\right)$  并用在专家系统 PROSPECTOR[24] 中来组合不确定的因素。

$$f(x) = \frac{\prod_{i=1}^n x_i}{\prod_{i=1}^n x_i + \prod_{i=1}^n (1 - x_i)}$$

其中约定  $0/0 = 0$ 。该函数在  $\left[0, \frac{1}{2}\right]^n$  上是合取的, 在  $\left[\frac{1}{2}, 1\right]^n$  上是析取的, 其他情况是平均的。该函数是关联的, 具有中立元素  $e = 1/2$ , 且在  $[0, 1]^n$  边界上是非连续的。

## 22.4 聚合函数的构建

有很多聚合函数。问题是如何选择一个最适合某一特定应用聚合函数。有时候一个函数就可以满足应用的所有组成, 在其他时候一个不同类型的聚合可能应用在各个阶段。下面的探讨可能会有所帮助。

### 22.4.1 数据收集和处理

数据类型以及数据收集的方式影响着方案的聚合。如果用户能够体贴地为连续范围内的每个物品提供精确的分数或者他们自己偏好的一定程度的确定性的数值描述, 那么 RS 将会非常轻易地做出相关的推荐。当然, 审美偏好通常局限在从用户那儿得到明确的信息, 因此需要加强交互体验。我们将简单地探讨一下系统能够获得的不同类型的数据, 以及这会对某些聚合函数的适合度产生怎样的影响。

**有序数据** 要求具体评分信息的 CF 推荐系统通常会做一个有限的有序量表。例如,  $\{1 = \text{不喜欢!}, \dots, 5 = \text{非常喜欢!}\}$ 。另一方面, 将用户行为转换为用户配置文件的有序部分是可行的, 例如,  $\{\text{定期观看}, \text{有时候观看等}\}$ 。只要有一个有序范围, 这些值就可以转换成数据以及被聚合。对于非齐次的和那些缺乏平移不变性特性的函数, 在单元区间内显示顺序值是有必要的。聚合值的不精确可能会使得加权算术平均和几何平均值被忽略。非

常适合有序数据聚合的聚合函数有 Sugeno 积分和诱导 OWA。

- Sugeno 积分  $Su$ (定义 22.9) 是一个可以处理有序数据并考虑交互的函数。模糊度量值应该与输入值处在同一个有序范围。Sugeno 积分可以模拟中位数函数以及最小化和最大化函数, 有将输出表示为有序值的优势。
- 诱导 OWA 函数[45]可以模拟最近邻方法即使相似度表示为有序值, 虽然它需要评分用数值表示。

**数值数据** 如果一个系统可以将用户输入和行为表示为数值数据, 考虑这些值是否精确是有用的, 不管他们是同样的还是独立的。像几何平均函数当输入值比算术平均的输入值高时变化率会更高。这有助于提供输出的粒度, 不过这也意味着此阶段的错误会影响推荐准确度。在 CF 中, 两个用户可能具有相同的偏好, 可是其中一个可能连续地高估物品。在这种情况下, 在聚合之前可能需要标准化一下评分, 这样用户之间的值就是可比的。使用 WAM 表示输入之间的独立性, 然而其他平均函数特别是 Choquet 积分, 可以表示某些输入或相关分数之间的交互或关系。

**分类数据** 有时候使用分类数据可能让聚合函数变得不可用。如果分类之间没有顺序, 使用平均或最大化和其他对建立用户间相似度有帮助的技术是无意义的。转换分类数据时可行的, 比如, 据其对 DF 中某个原型的贡献。

然而可能会有变化: 一些与  $d_i$  相关的向量的组成部分可能会丢失, 例如, CF 中的评分或者输入  $d_i = (x_1, \dots, x_n)$  可能会随着维度不同而变化。在其他情况下, 一些输入和输出的关联的不确定可能规定一个值的范围, 例如, 区间[6, 8]作为一部电影的评分。关联函数或生成函数可以根据属性来聚合具有一定连续性的不同维度的输入, 而数据转换或区间值函数可以用在后续的阶段。

## 22.4.2 期望属性、语义、解释

一旦知道了数据结构, 第一步是选择聚合函数通常要决定平均、合取、析取或混合需要哪一类。正如在 22.3.1.1 节中讨论的一样, 有时拥有一个可以将物品分类为偏好次序要比简单地提供可解释的输出重要得多。本章讨论 4 种聚合函数, 它们的语义可以用来决定需要哪类函数。

**最小化(合取的)** 最小化函数使用最小化输入作为输出。这意味着如果所有的输入都高, 那么该函数只能返回高输出。这种聚合对于使用式(22.5)的 UB 系统或者甚至需要所有输入都满足的 CB 系统是有用的。函数像结果(TP)对任何不完美输出有累积效应, 所以当维度较高时, 使用最小化函数会更有用。

**最大化(析取的)** 最小化函数像聚合函数一样模拟 AND 操作而析取函数模拟 OR 操作。这种类型的聚合产生的结果与最高输入相等或者比最高输入还要高。这在 KB、UB 和 CB 中也有用, 如果有多重偏好或标准一个好分数对于推荐方案足够。探讨例 22.7。

**例 22.7** 一个 CB 新闻推荐系统的用户与其配置文件相关的关键词有 {Haruki Murakami, X-Men, bushfires, mathematics, Jupiter orbit}, 任何一个新闻故事与所有甚至几个关键词具有很高相关性是不太可能的, 所以 RS 使用析取聚合作为推荐的基础。

**算术平均(平均的)** 当在 CF 中聚合用户评分和在 CB 中聚合物品特性时, 假设虽然评分很不相同, 但如果足够多的输入, 输出将会是可靠的。我们不希望推荐被一个孤立地对他所购买的不满意的用户或在 20 个能完全满足的特性中的一个特性而被严重影响。

**统一模型(混合的)** 在不同行为需要不同领域阶段的情况下，这就需要一个混合聚合函数。这可以直截了当地决定，只有所有输入的值包括所有的高取值才算是高，否则边界行为会影响函数的精确度。例如，统一模型的使用，它允许较高值提高分数也允许较低值拉低分数。一个有持续性高分的物品更喜欢一个具有几乎最高分数的函数而不是一个或两个低分数的函数。

聚合函数的某些属性可能使它们具有吸引力。表 22.1 列出了我们展示过的主要聚合函数以及它们是否经常或者在某些情况下满足在 22.4 节中详解的属性。

表 22.1 聚合函数及其属性

属性	$WAM_w$	$G_w$	$H_w$	$M_{w,[r]}$	$C_v$	$S_v$	$OWA_w$	max	min	$T_p$	$T_L$	$U$	$V$
幂等性	◆	◆	◆	◆	◆	◆	◆	◆	◆				
对称性	◇	◇	◇	◇	◇	◇	◆	◆	◆	◆	◆	◆	◆
不对称性	◇	◇	◇	◇	◇	◇							
关联性								◆	◆	◆	◆	◆	◆
严格单调	◇			◇	◇	◇	◇						
平移不变性	◆			◇	◆	◆	◆	◆	◆				
均匀性	◆	◆	◆	◆	◆	◆	◆	◆	◆				
利普希茨连续	◆		◆	◇	◆	◆	◆	◆	◆	◆	◆		△
中立性							◇	◆	◆	◆	◆	◆	
吸附性元素		◆	◆				◇	◆	◆	◆	◆		

◆=经常    ◇=依赖权重    △=依赖使用的  $T, S$

22.4.3 函数表现的复杂度及其理解

在某些情况下，简单的函数像 WAM 就可以满足推荐的目标了，但在别的方向有潜在的 RS 改进方法。根据其特性而言，WAM 是一个很健壮而通用的函数。它不偏向高或低的分数，也不会累积误差的影响，计算成本低，由于被广泛采用而更容易理解和解释。我们将幂平均和 Choquet 积分作为例子，它们的属性让它们更适合这种情况。

**幂平均** 幂平均是参数化函数，能够表示从最小化到最大化的渐进过程的函数，包括 WAM。在拟合技术可供我们使用时该函数立即生效，因为我们可以使用一个过程来确定任何数量的函数作为最佳候选。下面讨论调和平均数  $M_w$ 、 $[-1]$ 和均方值  $M_w[2]$ 。如果有一个输入为 0，那么调和平均数的输出就不能比 0 大。这为只允许考虑至少部分满足指标的物品提供了一个很好的解释，然而该函数不是合取的，所以仍然要给出一个介于最高分和最低分的一个分数。调和平均数也是凹的，且其输出对于任意  $d_i$  都要小于等于 WAM 的输出。对低输入的补偿可以减少，所以为了评分更高物品必须满足更多的指标。另一方面，二次幂平均更趋向于高分，有利于那些只有几个高分并且要为低分数特性和评分补偿更多的物品。

**Choquet 积分** 由于幂均值 Choquet 积分可以表示最小化和最大化函数之间的函数。Choquet 积分的使用最有趣的是在不对称的情况下，这些情况往往具有某些相关性。例如，在 KB 推荐系统中有时偏好会互相矛盾，而在其他时候又可能互相包含。在例子 Entrée[16]中，可以看到用户可能对一个价格便宜又不错的餐馆评分。因为通常会涉及一些权衡，所以满足这些指标的餐厅在推荐时应该格外奖励。在 CB 电影推荐系统中，一个用户可能喜欢 Johnny Depp 和 Tim Burton。如果有许多电影是 Tim Burton 导演并由

Johnny Depp 主演, 那么重复计算这些特性可能就没有意义了。Choquet 积分可以解释这些情况的组合, 因为每个指标子集都附有一个权重。“由 Depp 主演 Burton 导演”的子集的权重比各部分的权重总和小, 而 KB 中便宜又不错的餐馆的权重比较大。

当然, 有时数据结构可能很难理解和解释一个特殊函数的使用。在这种情况下, 有必要检查一下数据集上一些函数的精确性。进行最小化、最大化、算术平均和调和平均数的比较可能会显示哪些函数是有用的。

#### 22.4.4 权重和参数的确定

CF 中用来聚合评分的权重的确定通常是通过用户之间的相似性和邻居形式来推断的。CB 和 UB 中的权重是每个特性对用户的重要性的度量指标, 而加权 HS 中的权重表示的是推荐组成部分的可信度。权重可以通过预定措施, 如余弦来选择, 或者提前由 RS 设计者决定。例如, 我们决定使用权重向量  $w = (0.4, 0.3, 0.2, 0.1)$  为相似用户加权。一些系统通过与系统质量相关的隐性或显性反馈来递增地调整权重, 例如, P-Tango[19]中的混合 RS。在 22.5 节中我们将讨论确定可用数据集权重的方程。

### 22.5 推荐系统中的复杂聚合过程: 为特定应用定制

这里考虑协同过滤推荐系统中的拟合问题, 然而在基于内容和基于用户推荐系统中拟合权重也是可行的, 权重拟合的系统可以访问输入输出值, 因此拟合的强度可以确定权重或参数的适合度。拟合可以用在插值或近似值中。在插值情况下, 目标是精确地拟合特定的输出(在聚合函数中值对  $((0, 0, \dots, 0), 0)$  和  $((1, 1, \dots, 1), 1)$  应该一直是插值的)。在 RS 中数据经常会包含一些错误或近似度, 因此插值不准确的数值是不合适的。在这种情况下我们的目标是尽量接近期望输出而不是完全地匹配。这便是近似问题。

聚合函数的选择可以有如下描述表达:

给定一组数值属性  $P_1, P_2, \dots$  和数据集  $D = \{(x_k, y_k)\}_{k=1}^K$ 。选择在  $P_1, P_2, \dots$  上连续且满足  $f(x_k) \approx y_k, k=1, \dots, K$  的聚合函数。

我们也可以通过调整一个拟合区间来改变问题, 例如, 要求  $f(x_k) \in [\underline{y}_k, \bar{y}_k]$ 。这些值如何指定取决于具体应用。在某些情况下, 函数可能会精确地拟合且不会与所需的属性冲突, 然而大多数情况下我们需要最小化近似误差。

精确地说, 近似满足度的等式  $f(x_k) \approx y_k$  可以解释为下面的最小化问题:

$$\begin{aligned} & \text{minimize } \|r\| \\ & \text{subject to } f \text{ satisfies } \mathcal{P}_1, \mathcal{P}_2, \dots, \end{aligned} \quad (22.10)$$

其中  $\|r\|$  是残差的范数, 例如,  $r \in R_K$  是预测值与观察值  $r_k = f(x_k) - y_k$  之间的差异向量。有许多方法来选择范数, 最流行的是最小二乘范数

$$\|r\|_2 = \left( \sum_{k=1}^K r_k^2 \right)^{\frac{1}{2}}$$

和最小绝对偏差标准

$$\|r\|_1 = \sum_{k=1}^K |r_k|$$

或者是它们的加权类似, 如果  $y_k$  不可靠。

参照例 22.8<sup>①</sup>。

**例 22.8** 在 CF 推荐应用中我们想用五个相似用户为给定用户预测新物品的评分。现在有一个许多物品被用户评过分的物品的数据集以及五个相似用户或者邻居  $\{(d_i, R(u, d_i))\}_{i=1}^{10}$  其中  $d_i = (R(u_1, d_i), \dots, R(u_5, d_i))$  表示每个邻居  $u_1, \dots, u_5$  在过去物品  $d_i$  上的评分,  $R(u, d_i)$  表示用户真实的评分。例如, 前面的  $d_i = x_k, R(u, d_i) = y_k$ 。表 22.2 给出了一个拥有两个由邻居评分的物品的示例数据集, 其中这两个物品用户还没有评分而且可能会被推荐。我们想要通过一个为每个用户分配权重  $w_i$  的最小二乘法来定义一个加权算术平均方法。所以有如下公式:

$$\begin{aligned} & \text{minimize } \sum_{i=1}^1 0 \left( \sum_{j=1}^5 w_j R(u_j, d_i) - R(u, d_i) \right)^2 \\ & \text{subject to } \sum_{j=1}^5 w_j = 1 \\ & \quad w_1, \dots, w_5 \geq 0 \end{aligned}$$

这是一个二次规划问题, 可以用一系列的标准方法求解。在当前的例子中所得的模型权重分配为  $w = \langle 0.27, 0.07, 0.06, 0.19, 0.41 \rangle$ , 未评分物品的推荐分数为 4.7 和 7.9。观察与预测评分的最大差值为 2.5, 平均值为 0.98。如果我们使用余弦计算来确定权重, 则权重向量为  $w = \langle 0.19, 0.24, 0.23, 0.18, 0.17 \rangle$  推荐分数为 5.6 和 7.1。此方法的精确度与前者类似, 最大误差 2.48 和平均误差 1.6。有趣的是, 使用此方法时用户  $u_5$  是最不相似的用户, 但在为用户  $u$  进行精确预测时却是最重要的。◀

表 22.2 CF 中交互评分物品数据集示例

		用户及邻居对 $i=1 \cdots 10$ 的物品的评分										未评分	
用户评分	$R(u, d_i)$	6	4	6	8	10	5	7	7	5	5	?	?
邻居评分													
	$R(u_1, d_i)$	4	4	4	8	10	3	7	5	3	3	4	7
	$R(u_2, d_i)$	6	0	6	4	6	1	3	3	1	5	8	7
	$R(u_3, d_i)$	3	1	8	5	7	2	4	4	2	2	7	5
	$R(u_4, d_i)$	6	5	6	8	8	6	5	5	3	5	3	8
	$R(u_5, d_i)$	6	4	6	7	8	1	5	8	5	8	5	9

如上所提到的, 如果可供推荐的物品是有限的, 则排名显得格外重要(可参照[27]), 而不是预测精确度。在有意义的情况下, 结果的排名可以用  $f(R(u_1, d_k), \dots, R(u_n, d_k)) \leq f(R(u_1, d_l), \dots, R(u_n, d_l))$  if  $R(u, d_k) \leq R(u, d_l)$  对所有的值对  $k$  来存储, 其中  $l$  是额外的约束。在 CF 中, 对于这种情况, 相似用户的排名能更好地反映该用户的排名, 那么赋予该相似用户的权重就越高。这对一些用户可能会高估或者低估物品, 但对他们所偏好的物品保有一致性的情况有帮助。

到目前为止, 所讨论的近似问题被证明是一个基本的非线性优化问题, 或者一个特殊分类问题。一些优化问题使用凸目标函数或该函数的变体, 在这种情况下难的不是该步骤而是限制条件的定义。例如, 拟合 Choquet 积分就需要定义限制条件的指数量。许多问题都可以指定为线性或二次规划问题, 这些问题已被广泛研究并且已有许多可用的解决方

① 本节所有实例使用文献[8]中的软件包 aotool 和 ftools。



案。例 22.9 Choquet 积分作为期望函数使用相同的数据集(表 22.2)。在实际中, 鉴于 Choquet 积分定义在  $2^n$  个点上(所以理想情况下, 拟合的数据量应该远高于此), 所以最好拥有一个更大的数据集。这保证了结果函数不会太特化。

**例 22.9** (22.8 续) 系统设计者认为应该使用 Choquet 积分预测未知评分。为使拟合过程不易受离群点的影响, 他们决定使用最小绝对偏差标准, 并将优化过程表示为

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^5 |C_v(d_i) - R(u, d_i)| \\ \text{subject to} \quad & v(A) - v(B) \geq 0, \text{ for all } B \subseteq A \\ & v(A) \geq 0, \forall A \subset \mathcal{N}, v(\phi) = 0, v(\mathcal{N}) = 1 \end{aligned}$$

这使得 Choquet 积分可由一个有如下值的模糊度量定义:

$v(\{1\})=1, v(\{2\})=0.33, v(\{3\})=0, v(\{4\})=v(\{5\})=0.67, v(\{2, 3\})=0.33, v(\{2, 4\})=v(\{3, 4\})=v(\{3, 5\})=v(\{2, 3, 4\})=0.67$ , 所有其他子集有  $v(A)=1$ 。Shapley 值很好地表示了每个邻居的影响力, 由下式表示:

$$\phi_1=0.39, \phi_2=0.11, \phi_3=0, \phi_4=0.22, \phi_5=0.28$$

对于加权算术平均, 这些值显示邻居 1, 4, 5 可能与给定的用户更相似。我们还发现值对间的交互指数如下:

$$\begin{aligned} I_{12}=I_{24}=I_{45} &= -0.17, I_{14} = -0.33, I_{15} = -0.5 \\ I_{ij} &= 0 \quad (\text{其他对}) \end{aligned}$$

这明了一些邻居之间的冗余。特别是邻居 1 和 5 非常相似。该例的最大误差为 1.6, 平均误差为 0.6, 推荐分数为 6.0 和 8.7。由于代理变量, 该函数表现出与最大化函数的相似。后面的物品得分高, 主要是因为邻居 4 和 5 给出的高评分。

22.3.2 节定义的聚合函数成员用来理解和解释结果是很方便的。权重和参数有一个具体的意义, 而拟合这些函数主要涉及找到每个参数的最佳值来最大化 RS 的可靠性。

然而在其他情况下, 对某些事情的解释可能没有那么重要: 我们只是希望能够可靠并自动预测未知评分。有许多非参数化方法来构建聚合函数, 虽然没有系统解释的优势, 但可以自动构建并能很好地拟合数据。一个“黑盒”方法可以用来构建分段数据上的基本聚合操作。

我们可通过平滑数据以及确保每个分段都保持其特性来保证单调性和边界条件的指定。这里考虑基于样条聚合函数的构建[10]。

单调的张量积样条函数定义如下:

$$f_B(x_1, \dots, x_n) = \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \dots \sum_{j_n=1}^{J_n} c_{j_1 j_2 \dots j_n} B_{j_1}(x_1) B_{j_2}(x_2) \dots B_{j_n}(x_n)$$

如果希望构建的函数属于某个特殊的类或者具有某些特性, 在拟合时可以添加额外的限制条件。特别是我们可以通过表示具有系数  $c_{j_1 j_2 \dots j_n}$  的线性条件来保持单调性。此函数的拟合涉及稀疏矩阵, 矩阵的大小会随着基函数的变量个数而指数级增加。这里给出一个此拟合过程的例子, 例 22.10。

**例 22.10** (延续例 22.8~22.9) 我们的应用中不一定要知道相似用户的权重。我们仅想要自动构建可以预测未知物品的评分的函数。我们决定仍然需要单调性和幂等性来保证输出的可靠性, 并构建一个表示为张量积样条曲线的基本聚合函数。可以使用下面的二次规划问题:

$$\begin{aligned}
& \text{minimize} && \sum_{i=1}^5 (f_B(d_i) - R(u, d_i))^2 \\
& \text{subject to} && \sum_{j_1=1}^{J_1} \sum_{j_2=1}^{J_2} \cdots \sum_{j_n=1}^{J_n} c_{j_1 j_2 \cdots j_n} \geq 0 \\
& && f_B(0, \cdots, 0) = 0, f_B(1, \cdots, 1) = 0
\end{aligned}$$

幂等性通过使用多样插值条件得到保证, 例如,  $f_B(t_i, \cdots, t_i) = t_i$ 。这些条件必须以某种方式选择(可参照[6, 7])。拟合的非参数化函数给出未知物品的推荐结果分数 4.2 和 8.1, 所以看起来后者应该推荐给用户。

很明显, 使用非参数化还是参数化方法以及使用多么复杂的聚合函数是系统设计者的选择。推荐系统通常需要实时决定并处理大数据集, 所以需要寻找表达性和简单性之间的平衡。

## 22.6 总结

本章的目的是展示前端的聚合函数并介绍对推荐有益的函数的构建。这包括由许多权重定义的方法, Choquet 积分是由模糊度量的, 通过生成器构建的三角模型/三角余模型及可解释的统一模型。目前推荐系统使用的许多方法都涉及了构建加权算术平均值, 其中权重通过不同的相似性计算方法决定, 然而, 在许多情况下, 仅仅提高一点复杂度函数的精确度和可扩展性就可以得到改善。我们提供了详细的例子展现聚合函数用在推荐系统过程中的不同方法, 包括评分聚合、特性组合、相似性和邻居的形成以及加权混合系统中的组件组合。我们还在尝试找到能最好地模拟数据集的权重、相似性或者参数时引进了一些软件工具使得函数符合数据(可参考[29, 25])。

聚合函数的广泛研究并有大量的重要结果, 其中有些已经应用到了推荐系统中。由于这里提供的只是一个介绍, 我们推荐您阅读 22.7 节里所列的最近的书籍, 它们提供了许多聚合方法的详细内容。

## 22.7 进阶阅读

- Alsina, C., Frank, M.J. and Schweizer, B.: Associative Functions: Triangular Norms And Copulas. World Scientific, Singapore (2006)
- Beliakov, G., Pradera, A. and Calvo, T.: Aggregation Functions: A guide for practitioners. Springer, Heidelberg, Berlin, New York (2007)
- Calvo, T. and Mayor, G. and Mesiar, R.: Aggregation Operators: New Trends and Applications. Physica-Verlag, Heidelberg, New York (2002)
- Grabisch, M., Marichal, J.-L., Mesiar, R. and Pap, E.: Aggregation Functions. Cambridge University Press, Encyclopedia of Mathematics and its Applications, No 127, Cambridge (2009)
- Klement, E.P., Mesiar, R. and Pap, E.: Triangular Norms. Kluwer, Dordrecht, (2000)
- Torra, V. and Narukawa, Y.: Modeling Decisions. Information Fusion and Aggregation Operators. Springer, Berlin, Heidelberg (2007)

## 致谢

T. Calvo 想要感谢来自 Ministerio de Educaci' on y Ciencia、西班牙和欧洲项目 143423 - 2008-LLP-ES-KA3-KA3MP 的工程 MTM2006 - 08322 和 PR2007 - 0193 的支持。

## 参考文献

1. Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A.: Incorporating contextual information in recommender systems using a multi-dimensional approach, *ACM Transactions on information systems*, **23**(1), 103-145 (2005)
2. Adomavicius, G. and Kwon, Y.: New Recommendation Techniques for Multicriteria Rating Systems. *IEEE Intelligent Systems*. **22**(3), 48-55 (2007)
3. Ahn, H.J.: A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences*. **178**, 37-51 (2008)
4. Al-Shamri, M.Y.H. and Bharadwaj, K.K.: Fuzzy-genetic approach to recommender systems based on a novel hybrid user model. *Expert Systems with Applications*, **35**, 1386-1399 (2008)
5. Balabanovic, M. and Shoham, Y.: Fab: Content-Based, Collaborative Recommendation. *Comm. ACM*. **40**(3), 66-72 (1997)
6. Beliakov, G.: Monotone approximation of aggregation operators using least squares splines. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems*. **10**, 659-676 (2002)
7. Beliakov, G.: How to build aggregation operators from data? *Int. J. Intelligent Systems*. **18**, 903-923 (2003)
8. Beliakov, G.: FMTools package, version 1.0, <http://www.deakin.edu.au/~gleb/aotool.html>, (2007)
9. Beliakov, G. and Calvo, T.: Construction of Aggregation Operators With Noble Reinforcement. *IEEE Transactions on Fuzzy Systems*. **15**(6), 1209-1218 (2007)
10. Beliakov, G., Pradera, A. and Calvo, T.: *Aggregation Functions: A guide for practitioners*. Springer, Heidelberg, Berlin, New York (2007)
11. Beliakov, G., Calvo, T. and James, S.: On Lipschitz properties of generated aggregation functions. *Fuzzy Sets and Systems*, **161**, 1437-1447 (2010)
12. Beliakov, G. and James, S.: Using Choquet Integrals for kNN Approximation and Classification. In Gary G. Feng (ed.), 2008 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2008), 1311-1317 (2008)
13. Bolton, J., Gader, P. and Wilson, J.N.: Discrete Choquet Integral as a Distance Metric. *IEEE Trans. on Fuzzy Systems*, **16**(4), 1107-1110 (2008)
14. Buchanan, B. and Shortliffe, E.: *Rule-based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, MA (1984)
15. Bullen, P.S.: *Handbook of Means and Their Inequalities*. Kluwer, Dordrecht (2003)
16. Burke, R.: Hybrid Recommender Systems: Survey and Experiments, User Modeling and User-adapted interaction, **12**(4), 331-370 (2002)
17. Calvo, T., Kolesárová, A., Komorníková, M. and Mesiar, R.: Aggregation operators: properties, classes and construction methods. In : Calvo, T., Mayor, G. and Mesiar, R. (eds.) *Aggregation Operators. New Trends and Applications*, pp. 3-104. Physica-Verlag, Heidelberg, New York (2002)
18. Chen, Y.-L. and Cheng, L.-C.: A novel collaborative filtering approach for recommending ranked items. *Expert Systems with Applications*. **34**, 2396-2405 (2008)
19. Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D. and Sartin, M.: Combining Content-Based and Collaborative Filters in an Online Newspaper. In *Proceedings of SIGIR 99 Workshop on Recommender Systems: Algorithms and Evaluation*, Berkeley, CA (1999)
20. Campos, L.M.d., Fernández-Luna, J.M. and Huete, J.F.: A collaborative recommender system based on probabilistic inference from fuzzy observations. *Fuzzy Sets and Systems*. **159**, 1554-1576 (2008)
21. Dubois, D. and Prade, H.: *Fuzzy Sets and Systems: Theory and Applications*. Academic Press, New York (1980)
22. Dubois, D. and Prade, H.: *Fundamentals of Fuzzy Sets*. Kluwer, Boston (2000)
23. Dubois, D., Hillermeier, E., and Prade, H.: Fuzzy methods for case-based recommendation and decision support. *J. Intell. Inform. Systems*. **27**(2), 95-115 (2006)
24. Duda, R. Hart, P. and Nilsson, N.: *Subjective Bayesian methods for rule-based inference*

- systems. In Proc. Nat. Comput. Conf. (AFIPS), volume 45, 1075–1082 (1976)
25. Grabisch, M., Kojadinovic, I., and Meyer, P.: A review of methods for capacity identification in Choquet integral based multi-attribute utility theory: Applications of the Kappalab R package. *European Journal of Operational Research*, **186**, 766–785 (2008)
  26. Hibbert, R.: What is Indie Rock? *Popular Music and Society*, **28**(1), 55–77 (2005)
  27. Kaymak, U. and van Nauta Lemke, H.R.: Selecting an aggregation operator for fuzzy decision making. In 3rd IEEE Intl. Conf. on Fuzzy Systems, volume 2, 1418–1422 (1994)
  28. Klement, E.P., Mesiar, R. and Pap, E.: *Triangular Norms*. Kluwer, Dordrecht, (2000)
  29. Kojadinovic, I. and Grabisch, M.: Non additive measure and integral manipulation functions, R package version 0.2, <http://www.polytech.univ-nantes.fr/kappalab>, (2005)
  30. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., and Riedl, J.: GroupLens: applying collaborative filtering to Usenet news, *Communications of the ACM*, **40**(3), 77–87 (1997)
  31. Krawczyk, H., Knopa, R., Lipczynska, K., and Lipczynski, M.: Web-based Endoscopy Recommender System - ERS. In International Conference on Parallel Computing in Electrical Engineering (PARELEC'00), Quebec, Canada, 257–261 (2000)
  32. Linden, G., Smith, B., and York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, **7**(1), 76–80 (2003)
  33. Mayor, G., and Calvo, T.: Extended aggregation functions. In IFSA'97, volume 1, Prague, 281–285 (1997)
  34. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., and Riedl, J.: MovieLens unplugged: experiences with an occasionally connected recommender system. In Proceedings of the 8th international ACM conference on Intelligent user interfaces, Miami, USA, 263–266 (2003)
  35. Riedl, J. and Dourish, P.: Introduction to the special section on recommender systems, *ACM Transactions on Computer-Human Interaction*, **12**(9), 371–373 (2005)
  36. Resnick, P., Iakovou, N., Sushak, M., Bergstrom, P., and Riedl, J.: GroupLens: An open architecture for collaborative filtering of Netnews. In Proceedings of ACM conference on computer supported cooperative work, Chapel Hill, NC, 175–186 (1994)
  37. Rokach, L., Maimon, O., and Arbel, R.: Selective voting-getting more for less in sensor fusion, *International Journal of Pattern Recognition and Artificial Intelligence* **20** (3), pp. 329–350 (2006)
  38. Rokach, L. and Maimon, O.: Theory and applications of attribute decomposition, *IEEE International Conference on Data Mining*, IEEE Computer Society Press, pp. 473–480 (2001).
  39. Salton, G. and McGill, M.: *Introduction to Modern Information retrieval*. McGraw Hill, New York (1983)
  40. Schafer, J.B., Konstan, J.A., and Riedl, J.: E-commerce recommendation applications, *Data Mining and Knowledge Discover*, **5**, 115–153 (2001)
  41. Santini, S. and Jain, R.: Similarity Measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**(9), 871–883 (1999)
  42. Ujjiin, S., and Bentley, P.: Using evolutionary to learn user preferences. In: Tan, K., Lim, M., Yao, X. and Wang, L. (eds.). *Recent advances in simulated evolution and learning*, pp. 20–40. World Scientific Publishing (2004)
  43. Yager, R.: On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Trans. on Systems, Man and Cybernetics*, **18**, 183–190 (1988)
  44. Yager, R.: Noble Reinforcement in Disjunctive Aggregation Operators. *IEEE Transactions on Fuzzy Systems*, **11**(6) 754–767 (2003)
  45. Yager, R. R. and Filev, D. P.: Induced ordered weighted averaging operators. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, **20**(2), 141–150 (1999)
  46. Yager, R. and Rybalov, A.: Uninorm aggregation operators. *Fuzzy Sets and Systems*, **80**, 111–120 (1996)

## 推荐系统中的主动学习

Neil Rubens、Dain Kaplan 和 Masashi Sugiyama

### 23.1 简介

通常假设推荐系统向用户展示物品的原因只是要向用户推荐他可能感兴趣的物品。推荐系统当然要推荐物品,但这种假设虽然与字面上的称谓无关,却曲解了系统在做的“推荐”。为用户展示一个物品还有另一个原因:更多地了解他/她的喜好,或者他/她的好恶。这就是主动学习(Active Learning, AL)。为推荐系统增加主动学习功能能够帮助用户更好地意识到自己的好恶,同时可以为系统提供新信息以便分析后续的推荐。其实,在推荐系统中引入主动学习能够个性化推荐过程,该过程是有意义的,因为推荐本质上就是个性化。该过程使得系统主动改变展示给用户哪些物品(例如,用户在注册或正常使用情况下的物品展示),并让用户能够自由地发掘他们的兴趣。

可惜系统鲜有机会能够在用户评论/阅读物品时或通过用户的浏览历史获取相关信息<sup>⊖</sup>。由于这种机会不多,所以我们希望尽可能确保这些数据能够告诉我们关于用户偏好的重要信息。毕竟,用户数据是一个公司的最有价值的资产之一。

例如,当一个新用户开始使用推荐系统时,系统对其偏好所知甚少[44, 36, 2]。获取用户偏好的一个普通方法是要求用户对一组物品评分(被称为训练集合)。于是估计用户偏好的模型应运而生。由于用户浏览的物品数量不能覆盖全部类别(有可能使得主动学习任务与推荐做无用功),那么供用户浏览的物品集合一定是非常有限的。因此学习模型的准确性很大程度上依赖于良好的训练集合的选择。系统可能会要求用户为星球大战 I, II, III 进行评分。通过用户对三部曲的评分,我们将很好地了解用户对星球大战的偏好,推而广之,了解用户对其他科幻电影的爱好,当然总体上收集到的信息是有限的。所以选择该三部曲并不能提供丰富的信息<sup>⊖</sup>。选择类似星球大战的流行物品的另外一个问题,就是根据定义大部分人会喜欢它们(否则就不流行了)。那么选择流行物品只能获得很少的用户信息(除非用户的品味特殊),就不足为奇了。

有这样一个观点认为主动学习是一个令人讨厌具有侵入性的过程,但它并不必如此[54, 38]。如果展示的物品用户感兴趣,那么这有可能是一个发现与探索的过程。有的推荐系统提供一个“给我惊喜!”的按钮来激励用户进入探索过程,而且确实有许多毫无购买意图的用户浏览看看有什么建议。探索是使用户更好地意识到自己偏好(是否改

---

Neil Rubens, University of Electro-Communications, Tokyo, Japan, e-mail: rubens@hrstc.org

Dain Kaplan, Tokyo Institute of Technology, Tokyo, Japan e-mail: dain@cl.cs.titech.ac.jp

Masashi Sugiyama, Tokyo Institute of Technology, Tokyo, Japan e-mail: sugi@cs.titech.ac.jp

翻译:北京大学-王雪丽 审核:刘良良

⊖ 利用社交网络获取额外数据是日益增长的趋势(参见第 18 章和第 19 章)。

⊖ 除非我们的目标是学习某种小偏好,即喜欢的类别里倾向“更挑剔”某些可选物品。



变), 同时将偏好告知系统的关键。请记住, 在某种意义上用户不仅可以通过他们对物品的评分, 也可以通过他们所消费的物品进行定义, 所以促使用户为不同的物品评分有可能进一步区分彼此的喜好, 并使得系统能够提供更好的个性化服务和更好地满足他们的需求。

本章只是对推进系统中主动学习的一个浅析<sup>⊖</sup>。然而我们希望本章能够提供必要的基础。

如想进一步阅读, 文献[46]对机器学习(侧重于自然语言处理和生物信息学)情景下的主动学习做出了一个不错的整体介绍。对关于主动学习的理论研究(主要专注于实验设计领域)可以参照文献[7, 4, 22]; 还有计算机科学领域中最近的研究成果[16, 5, 51]。

### 23.1.1 推荐系统中主动学习的目标

不同的推荐系统有不同的学习目标(见第 8 章), 使得它们的主动学习部分也需要不同的目标。因此, 一个主动学习方法可能比另外一个方法更能满足某个给定的任务[35]。例如, 正在构建的推荐系统中什么是重要的(见第 10 章)? 注册的难度(用户负担)? 用户对服务是否满意(用户满意度)? 系统预测用户偏好的功能如何(准确度)? 系统是否能很好地表达用户的偏好(用户效用)? 系统如何将从一个用户学习到的信息应用在其他用户上(系统效用)? 系统的功能性也可能比较重要, 例如, 当一个用户在查询感兴趣的物品的评分时, 系统没有足够的数据来预测, 这时系统会如何响应? 在这种情况下, 是否会给用户一个模糊的答案, 从而允许用户在有兴趣和时间时对系统进行进一步的训练? 还是要求他们在系统提供预测之前对其他物品进行评分? 或许用户之前体验过该物品(例如, 看过这部电影或者预告片)并认为他们的评分与预测评分相差很大[10]。在所有情况下系统如何响应用户值得仔细思考。

一般主动学习不考虑探索(学习用户偏好)与开发(利用用户偏好)的权衡, 也就是说它并不根据系统目标动态分配探索/开发的权重。而这种权衡是很重要的, 因为对于一无所知或所知甚少的新用户来说, 通过为用户提供可能感兴趣的预测(开发)来验证系统的价值是有益的, 同时长期用户希望能够通过探索来扩展自己的兴趣[38, 41]。

虽然一个推荐系统的目标可能是为用户提供准确的预测, 但系统也可能需要推荐高新颖性/惊喜度的物品, 提高覆盖率, 使利润最大化或者决定用户是否能够评估给定的物品, 举几个例子[44, 21, 33]。多重目标可能需要同时加入考虑(见第 24 章), 例如, 最小化训练数据净购置成本的同时最大化净利润, 或者找到为用户提供物品, 期望输出与效用和无用的替代效用之间的最佳搭配[38]。训练的效用也可能是很重要的, 例如, 为用户提供他们负担不起的进口车的预测评分可能是毫无用处的, 所以应该避免这种情况。可以看出系统目标常常要比单纯的预测准确度复杂得多, 并且可能包含多个目标的结合。

而推荐系统往往有一个不明确或开放式的目标, 即预测用户可能感兴趣的物品, 正如其名, 基于对话的主动学习[32, 37, 9]将与用户对话作为面向目标方法。它通过问题的迭代引出用户响应来减少在快速发现用户寻找内容时的搜索空间(见 23.8 节)。

**新用户问题** 当用户刚开始使用一个推荐系统时, 他们希望通过少量训练就可以看到有趣的结果。尽管系统对用户偏好所知甚少, 但选择能够最大化理解新用户需求的训练集

⊖ 有关主动学习的补充材料请访问: <http://www.DataMilking.org/AL>。

合进行评分是很重要的[35]。

**新物品问题** 随着新物品的引进,通过选择用户对其评分来快速提高这些物品的预测准确性是很重要的[24]。

**获取输出值的成本** 获取输出值的不同方法有着不同的成本。例如,将用户点击一个推荐物品作为正输出、不点击作为负输出的内因策略的用户负担很小。相反,要求用户明确地为物品评分成本比较高,即使这取决于任务。看类似星球大战的电影后来评分也许会提供不错的结果,但是需要大量的用户负担[20],而对笑话进行评分就不需要。这通常与探索/开发耦合不谋而合,且从不同输入获取输出的权衡也应该考虑在内(如确定性/不确定性、易于评估性等)。

**适应不同的主动学习方法** 虽然我们一直专注于减少预测错误的传统目标,同样构建一个方法来最大化其他目标如利润也是合理的。在这种情况下模型将选用增加利润而不是评分准确度的集合。

### 23.1.2 例证

让我们看看推荐系统中主动学习的一个具体例子。这里只阐述了一些概念,所以有些过于简单。请记住相似度度量方式会随着使用的方法不同而不同。这里假设如果电影属于相同的类型那么他们彼此相似。图 23.1 显示两个图表,最左边的是起始状态,其中右上方用户组已对一部电影评分,该电影属于科幻类。右边的图表显示了 4 个下一步可供选择的训练集合(a)、(b)、(c)或(d)。如果我们选择了训练集合(a)一个模糊类型电影(如金鱼缸人)并不会影响预测结果,因为邻域内没有其他电影(集合)。如果我们选择训练集合(b),我们就可以为相同区域的集合进行预测,但是这些预测可能已经从相同区域内的训练集合获得(根据左边的图表)。如果选择了训练集合(c),我们可以进行新的预测,但只可以对该区域内的其他三个集合,恰好是僵尸电影。选择训练集(d)可以对该区域的大量测试集合进行预测,预测结果为喜剧电影。因此(d)是最理想的选择,因为这允许我们对预测结果做出最大的改进(因其具有最多的训练集)<sup>⊖</sup>。

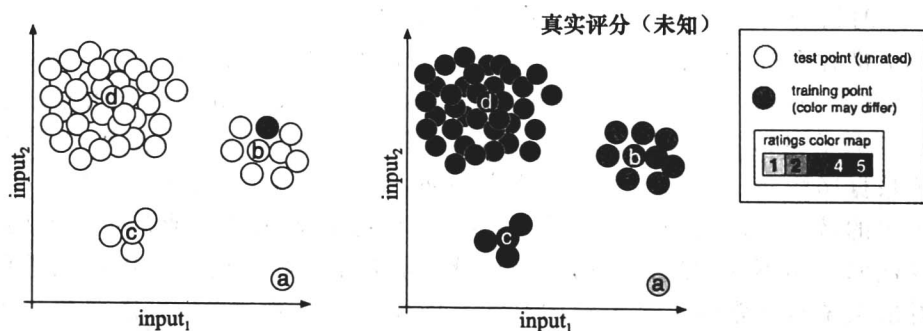


图 23.1 主动学习:例证(见 23.1.2 节)

### 23.1.3 主动学习的类型

本章的主动学习方法分类基于我们对它们的主要动机/目标的解释。需要注意的是,

<sup>⊖</sup> 这可能取决于推荐系统中采用的特定预测方法。

给定的主动学习方法可能有很多种分类方法,例如,接近于某个决策边界的取样可能会被归类为基于不确定输出的方法,因其输出未知;基于参数的方法,因其会改变模型;或者基于决策边界的方法,因其边界线会因此偏移。然而,由于该取样的执行与决策边界有关,我们将考虑该方法的首要动机,并根据该方法分类。

除了通过主要动机进行分类(见 23.1 节),为了便于理解,进一步将一个方法的算法细分为两个常见的分类类型:基于实例的和基于模型的。

**基于实例的方法** 该类型的方法根据它们的属性选择集合,试图在没有显性知识的基础模型中通过找到系统中与其他用户最匹配的用户来预测评分。对于此类型的通用名称包括基于内存的、惰性学习、基于实例的和非参数化方法[2]。假设现有的任何数据以及根据基础模型得到的评分都是可读取的。

**基于模型的方法** 该类型的方法选择集合来尝试构建一个最好的模型,该模型可以解释用户提供的用来预测用户评分的数据[2]。这些集合也用来最大化地减少模型的预测错误。这里假设除了基于实例的方法可访问的所有数据,还可以访问模型及其参数。

**自主学习模式:批处理和顺序** 处理因为用户通常希望系统可以立即输出一些有趣的东西,所以一个常用的方法是在用户为一个物品评分后以顺序的方式计算该用户的预测评分。同时也可以调整模型之前允许用户对多个物品或一个物品的多个特性进行评分。另一方面,顺序的选择训练集有个好处,即允许系统响应用户提供的数据并立即做出必要的调整。尽管该方法需要付出每一步都需要与用户交互的代价。由此批处理主动学习与顺序主动学习二者之间产生了一个权衡:数据利用率与用户交互的次数。

## 23.2 数据集的属性

在考虑任何主动学习方法时,为了最大化给定集合的效用,以下三个因素应该考虑在内。前两个因素的补充说明如下。以例证为例(图 23.1)。

(R1)被表示的:已经被现有的训练集表示了吗?如集合(b)。

(R2)可表示的:该集合是一个表示其他数据集的合适候选吗?或者是一个离群点吗?如集合(a)。

(R3)结果:选择该集合会产生更好的预测评分或者完成另一个目标吗?如集合(d)甚至集合(c)。

**(R1)由训练数据表示** 正如本章的简介中所解释的,要求对一个三部曲如《星球大战》的若干卷进行评分并不一定有好处,因为对于获取用户偏好新信息可能贡献不大。因此为了避免获得冗余信息,主动学习方法应该支持训练集中未被完全表示的数据[18]。

**(R2)测试数据的代表** 有一点很重要,就是任何通过主动学习算法选择进行评分的物品要尽可能地代表测试物品集(这里认为所有物品都可能属于测试集),因为算法的准确度将基于这些物品进行评估。如果被选电影来自一个小的类别,像例证(图 23.1)中的僵尸电影,那么获取该电影的评分对其他主流类别的用户偏好贡献不大。另外用户通常倾向于对自己感兴趣的电影类型评分,这就意味着任何在训练集中占优势的类别可能只代表了所有物品的一小部分[38]。为了增加信息获取量,选择能够提供关于其他未被评分物品的信息的具有代表性的物品是至关重要的[18, 47, 53]。

### 23.2.1 其他考虑因素

除了 23.2 节列出的三个条件,数据集的其他标准也应该被考虑,如下面的标准。

**代价** 正如本章简介所说,从用户选择中获取隐式反馈要比要求用户显示地对物品评分代价小[19]。这可看作一个可变成本问题。处理该问题的一个方法是考虑为物品打标签的成本以及将来评估加入训练集物品被错误分类的代价[27]。然而成本有可能是事先未知的[48]。

**可估价性** 用户有可能不会提供物品评分,你不可能为一部没看过的电影正确评分!因此建议也要考虑用户能够评估一个物品的概率[20]。

**显著性** 以决策为中心的主动学习强调评分更能够影响决策的物品,并且需要与决策相关的实例,这样一个相对较小的评估变化能够改变预测评分顶部的物品顺序[43]。例如,无用的标记物品会导致用户首页推荐的前十部电影(突出物品)的排名移动或重新排列,它可能被认为没多大用处。这也可能出现在只考虑获取系统强烈推荐的物品评分的情况下[6]。

**流行性** 建议将物品的流行性也考虑在内[35],例如,有多少人对该物品进行了评分。该操作基于这样一个理念,因为流行物品有许多人评分,所以它有可能信息相当丰富。相反地,也应该考虑一个物品的评分不确定性,因大多数用户趋向于为积极物品打高分,这表明该物品并不能提供太多的辨别力,因而不值得包括在训练集中。

**最佳/最差** 文献[29]中已指出查看最佳/最差评论,在用户对一个物品做出决定时是有用处的。将此观点扩展到主动学习中,我们假设这样一个“最佳/最差”原则,获取用户的最佳/最差评分(因其可能很好地捕获用户的偏好)对于用户偏好做出决定是有益的。通过要求用户提供他/她最喜欢/不喜欢的物品将主动学习问题转化为要求用户对已知分类中的物品进行评分的问题(例如,在喜欢的类别中选择一部喜欢的电影),以及获取物品过程招致的成本问题[30]。这个过程被称为“主动类选择”。该过程与传统主动学习技术相反,传统技术中标记物品的过程(而不是物品本身)被认为是产生成本的原因。

### 23.3 主动学习在推荐系统中的应用

在传统的主动学习中用户被要求对预选的物品评分。这往往是在刚加入的时候,虽然预选的列表在以后也会展示给已存在的用户。有人可能认为既然这些物品是由专家挑选的,那么它们应该已经捕获了用来决定用户偏好的重要属性。从概念上可能听起来比较合理,但是实际中这会导致选择的物品只能最好地预测用户平均的偏好。由于推荐系统的思想是提供个性化推荐,所以通过个性化方法选择评分物品应该更有意义。

#### 23.3.1 方法总结矩阵

下面的矩阵(表 23.1)总结了本章概述的方法。未提供明确的性能数字,据我们所知还不存在这样的综合比较。主动学习方法比较时可能带有个人偏见,任何结果都将是不确定的。这是因为作者有固定预测方法,然后将其应用在一个或多个主动学习方法中比较性能的倾向。然而,主动学习方法经常是为某个特定的预测方法设计,因此若应用在不同的预测方法中性能可能并不好(这会产生潜在的误导性结果),或者如果基础系统不能提供所需信息,这些方法甚至不可用,如评分评估分布。以上原因使我们不得不忽略任何形式的性能数字。

表 23.1 方法总结矩阵

方法主要动机	描述/目标	可能的限制条件
降低不确定性(见 23.5 节)	减少以下内容的不确定性： <ul style="list-style-type: none"> <li>• 评分评估(见 23.5.1 节)</li> <li>• 决策边界(见 23.5.2 节)</li> <li>• 模型参数(见 23.5.3 节)</li> </ul>	降低不确定性可能并不能总是提高准确率；该模型可能只对错误的事情肯定(如用错了预测方法)
减少错误(见 23.6 节)	通过错误与以下内容的关系来减少预测错误： <ul style="list-style-type: none"> <li>• 输出评估中的变化(见 23.6.1.1 节)</li> <li>• 测试集错误(见 23.6.1.2 节)</li> <li>• 参数评估中的变化(见 23.6.2.1 节)</li> <li>• 参数估计值的方差(见 23.6.2.2 节)</li> </ul>	可靠地评估错误的减少是困难并且计算昂贵的
基于聚类(见 23.7 节)	基于以下内容之间的共识找出有用的训练集： <ul style="list-style-type: none"> <li>• 聚类中的模型(见 23.7.1 节)</li> <li>• 多重候选模型(见 23.7.1 节)</li> </ul>	由于其效用取决于模型/候选的质量且可以是计算昂贵的，因为它的执行与多重模型/候选有关

## 23.4 主动学习公式

被动学习(见图 23.2)是指训练数据预先提供或系统无须努力获取新数据(仅通过用户行为随着时间不断积累)。

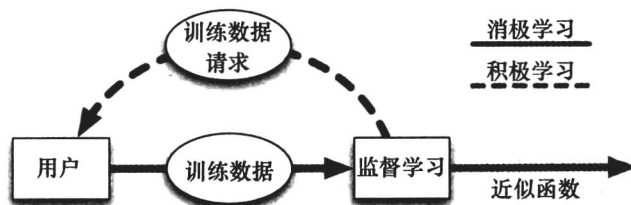


图 23.2 主动学习应用一个交互/迭代的过程来获得训练数据，不像被动学习，数据是给定的

另一方面主动学习主动地选择训练集(输入)以便观察信息最丰富的输出(用户评分行为等)。

让我们用一个更正式的方式来定义主动学习问题。一个物品被看作一个多维输入变量并用一个向量  $\mathbf{x}$  描述(也称为一个数据点)<sup>⊖</sup>。整个物品集表示为  $\mathcal{X}$ 。用户  $u$  的偏好表示为函数  $f_u$ (也称为目标函数)；为了简洁起见，用  $f$  指代目标用户。物品  $\mathbf{x}$  的评分被当作一个输出值(或者标签)用  $y = f(\mathbf{x})$  来表示。每个物品  $\mathbf{x}$  都可以在一个有限的范围  $\mathcal{Y} = \{1, 2, \dots, 5\}$  上进行评分。

在监督学习中，物品和相应的用户评分通常被分为互补的两部分——训练集合测试集(也称为验证集)。监督学习的任务就变成了，在给定一个训练集(通常由所有用户的评分提供)上学习一个准确预测用户偏好的函数  $\hat{f}$ 。训练集的物品表示为  $\mathcal{X}^{(\text{Train})}$ ，这些物品以及它们相对应的评分构成一个训练集，例如  $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_{\mathbf{x}_i \in \mathcal{X}^{(\text{Train})}}$ 。我们通过泛化误差来衡量学习函数预测用户真实偏好的准确度：

⊖ 表示物品的方法取决于推荐系统和基本的预测方法。在基于协同过滤的方法中，物品可能通过用户的评分来表示，或者在基于内容的推荐系统中，物品由它们的描述信息来表示。



$$G(\hat{f}) = \sum_{x \in \mathcal{X}} \mathcal{L}(f(x), \hat{f}(x)) P(x) \tag{23.1}$$

然而在实际中，并不是所有  $x \in \mathcal{X}$  都有  $f(x)$ ，因此常用测试误差来近似泛化误差：

$$\hat{G}(\hat{f}) = \sum_{x \in \mathcal{X}^{(\text{Test})}} \mathcal{L}(f(x), \hat{f}(x)) P(x) \tag{23.2}$$

其中  $\mathcal{X}^{(\text{Test})}$  是指测试集物品，预测误差通过利用一个损失函数  $\mathcal{L}$  来衡量，如绝对平均值 (MAE)：

$$\mathcal{L}_{\text{MAE}}(f(x), \hat{f}(x)) = |f(x) - \hat{f}(x)| \tag{23.3}$$

或者均方误差 (MSE)：

$$\mathcal{L}_{\text{MSE}}(f(x), \hat{f}(x)) = (f(x) - \hat{f}(x))^2 \tag{23.4}$$

主动学习标准被定义用来评估获取物品  $x$  的评分效用，并将其添加到训练集  $\mathcal{X}^{(\text{Train})}$  以获取某一特定目标 (见 23.1.1 节)。为简单起见，将此目标设为学习函数在训练集上的最小化泛化误差。我们将主动学习准则表示为

$$|\hat{G}(\mathcal{X}^{(\text{Train})} \cup \{x\})| \tag{23.5}$$

或为简单起见表示为

$$\hat{G}(x) \tag{23.6}$$

主动学习的目标为选择一个能够最小化泛化误差  $\hat{G}(x)$  的物品  $x$ ：

$$\operatorname{argmin}_x \hat{G}(x) \tag{23.7}$$

如果要求用户对  $x_j$  或物品  $x_k$  进行打分，那么可以通过一个主动学习标准来评估它们的效用，例如， $\hat{G}(x_j)$  和  $\hat{G}(x_k)$ ，选一个可以使得泛化误差更小的物品。注意我们需要在未知一个物品的真实评分时评估一个用户的评分。为区别候选评分物品与其他物品，我们将其表示为  $x_a$ 。只要提供所需信息，主动学习可以应用到任何预测方法中，例如，评分评估 [42] 以及它们的分布 [23, 25]、近似决策边界 [55, 15]、方法参数 [49] 等。

$x$	单个输入 (单个物品)
$\mathcal{X}$	多个输入 (多个物品)
$y$	输出 (物品的评分)
$\mathcal{Y} = \{1, 2, \dots, 5\}$	可能的输出 (评分) 例 $y \in \mathcal{Y}$
$f$	用户的偏好函数 (系统未知)
$\mathcal{X}^{\text{Train}}$	训练集输入 (已被评分的物品)
$\mathcal{T} = \{(x_i, y_i)   x_i \in \mathcal{X}^{(\text{Train})}\}$	训练集 (物品及其评分)
$\hat{f}$	用户偏好的近似函数 (来自训练集)
$G$	泛化误差 (预测准确度)，见 (23.1 节)
$x_a$	用来评分的物品
$\hat{G}(x_a)$	主动学习标准 (评估物品 $x_a$ 评分的效用)

图 23.3 符号总结

**回归和分类** 预测用户评分问题可以看作一个回归或分类问题。因为评分是离散的数值，所以是一个回归问题，例如，如果考虑它们的顺序属性，意味着评分可以排序 (如评分 4 比评分 3 高)。另外可以忽略评分的数值属性，通过将评分看作类/标签，把该问题当作一个分类问题<sup>⊖</sup>。例如，可以使用最近邻 (NN) 方法来分类，如从邻居中提取频率最高

⊖ 如果考虑标签的顺序属性，则称为有序分类。

的标签, 或者还可以用 NN 进行回归, 如计算邻居评分的平均值。在本章使用分类和回归的例子, 选择其中最合适的帮助解释。

## 23.5 基于不确定性的主动学习

基于不确定性的主动学习尝试获取训练点来减少某些方面的不确定性, 例如, 与输出值相关[28]、模型参数[23]和决策边界[45]等。该方法的一个潜在缺点是降低不确定性并不总是有效。如果一个系统能够确定用户评分, 这并不意味着一定是准确的, 因为有可能只确定错误的内容(例如, 如果算法错了, 减少再多不确定性也毫无帮助)。举个例子, 如果用户到目前为止评分一直比较正面, 那么系统可能误以为该用户喜欢所有的物品, 这可能很不正确。

### 23.5.1 输出不确定性

在基于输出不确定性的方法中, 标记选择的物品(训练集)来减少训练物品的预测评分不确定性。在图 23.1 中, 假设推荐系统在物品归属类的基础上评估物品的评分(例如, 相同类别的电影得分相同), 如果已经得到了某用户对一部科幻类(右上方)电影的评分, 那么推荐系统很有可能更加确定其他科幻类电影的评分, 并有利于获取用户对还未抽样的某个类别(聚类)中电影的评分, 如还未确定的聚类。

基于实例和模型的输出不确定性主动学习方法之间的主要不同之处在于, 对于任一项  $x$ , 评分分布函数  $P(Y_x)$  是如何获得的, 其中评分分布是指某个物品被赋值为某个评分的概率。基于模型的方法可以通过模型获取评分分布。由于概率模型直接提供评分分布, 所以特别适合这种情况[23, 25]。

对于基于实例的方法, 通过收集数据来获取评分分布。例如, 使用最近邻技术的方法在邻居投票的基础上获取评分分布, 其中这里的“邻居”是指具有相似偏好的用户, 公式如下:

$$P(Y_x = y) = \frac{\sum_{m \in NN_{x,y}} w_m}{\sum_{m \in NN_x} w_m} \quad (23.8)$$

其中,  $NN_x$  是为物品  $x$  打过的分的邻居,  $NN_{x,y}$  是给定物品  $x$  评分为  $y$  的邻居,  $w_m$  为邻居的权重(如相似度)。

#### 23.5.1.1 主动学习方法

一些主动学习方法[28]通过计算输出值的不确定性的本地(贪婪)方法来评估一个潜在训练集的效用:

$$\hat{G}_{\text{Uncertainty}_{\text{local}}}(\mathbf{X}_a) = -\text{Uncertainty}(Y_a) \quad (23.9)$$

由于我们的目标是最小化  $\hat{G}$ , 那么对物品评分具有较高不确定性是有利的, 这将消除被选物品评分的不确定性。但标记评分不确定的物品并不能达到减少其他物品评分不确定性的目标(例如, 标记离群点只会减少其他类似物品的不确定性, 就像表 23.1 中, 选择僵尸类型的物品(c)一样, 或者甚至在(d)中什么也没选)。

因此我们可能要考虑通过选择能够降低其他未评分物品不确定性的全局方式来降低不确定性。这样做的一种方法[40]是, 通过在整个测试集  $\mathcal{X}^{(\text{Test})}$  上相对于潜在训练物品  $x_a$  计算评分的不确定性来定义规则:

$$\hat{G}_{\text{Uncertainty}}(\mathbf{x}_a) = \frac{1}{|\mathcal{X}^{(\text{Test})}|} \sum_{\mathbf{x} \in \mathcal{X}^{(\text{Test})}} \mathbf{E}_{\mathcal{T}(a)}(\text{Uncertainty}(Y_{\mathbf{x}})) \quad (23.10)$$

其中  $\frac{1}{|\mathcal{X}^{(\text{Test})}|}$  归一化因子,  $\mathbf{E}_{\mathcal{T}(a)}(\text{Uncertainty}(Y_{\mathbf{x}}))$  表示不确定性的期望值, 其中将候选物品  $\mathbf{x}_a$  的评估评分  $y_a$  加入训练集  $\mathcal{T}$ , 如  $\mathcal{T}(a) = \mathcal{T} \cup (\mathbf{x}_a, y_a)$ 。

非本地化方法有一个潜在的缺点, 即对于本地化方法只需要评估单个输出值  $y_a$  的不确定性, 而对于非本地化方法则需要评估对潜在训练集  $(\mathbf{x}_a, y_a)$  的所有训练集输出值的不确定性。这有可能难以正确地评估并且计算昂贵。

### 23.5.1.2 不确定性度量

物品评分(输出值)的不确定性通常通过其方差、熵[28]或者置信区间[38]来计算。当评分最大程度偏离平均值时方差最大, 当所有评分差不多时熵被最大化。

输出值的不确定性可以用下式中的方差定义来计算:

$$\text{Uncertainty}(Y_a) = \text{VAR}(Y_a) = \sum_{y \in \mathcal{Y}} (y - \bar{Y}_a)^2 P(Y_a = y) \quad (23.11)$$

其中  $\bar{Y}_a$  表示物品  $\mathbf{x}_a$  上所有用户的平均评分,  $P(Y_a = y)$  是物品评分  $Y_a$  与  $y$  相等的概率, 二者要么是在基于实例的最近邻方法的基础上进行计算的, 要么是从基于模型方法的模型中获取的。

不确定性也可以由如下式的熵进行计算:

$$\text{Uncertainty}(Y_a) = \text{ENT}(Y_a) = - \sum_{y \in \mathcal{Y}} P(Y_a = y) \log P(Y_a = y) \quad (23.12)$$

文献[47]中提出一个基于最相似评分概率计算评分不确定性的方法:

$$\text{Uncertainty}(Y_a) = -P(Y_a = y^*) \quad (23.13)$$

其中  $y^* = \arg\max_y P(Y_a = y)$  是最相似评分。

在文献[38]中置信区间在计算选择训练输入集时用来计算不确定性:

$$c = P(b_l(Y_a) < y_a < b_u(Y_a)) \quad (23.14)$$

其中  $c$  是实际评分  $y_a$  位于下界  $b_l(Y_a)$  和上界  $b_u(Y_a)$  之间的置信度。例如, 系统可以确定物品有  $c=90\%$  的概率被评分为 3~5。许多方法偏爱上界较高的物品, 这表明该物品可能评分比较高(对开发有益), 而且如果置信区间也比较大, 那么也有可能利于探索。在某些时候, 理想的情况是增加预测评分比较高的物品数量, 在置信区间的下界使用期望值有利于物品选择, 期望值越高情况越理想。

### 23.5.2 决策边界不确定性

在基于决策边界的方法中, 选择训练集来改善决策边界。通常现有的决策边界被认为有几分正确, 所以采样接近决策边界的点以进一步修正(图 23.4)。从某种程度上说这也可以被认为是基于输出不确定性, 因为接近决策边界的集合的不确定性可能比较高。该方法在这样的假设上进行, 基础学习方法(如支持向量机)的决策边界很容易得到。该方法的一个明显优势是给定一个决策边界, 通过近似决策边界来选择训练样本的计算开销小。

正如文献[45]中所讨论的一样, 通过选择训练集来获取更准确的分割超平面(图 23.4b), 或如果超平面的方向已经确定, 可能要选择输入集来降低边缘的大小(图 23.4c)。虽然在靠近决策边界抽取训练集比较好是显而易见的[55, 15], 但是也有其他方法是选择远离边界的物品[15], 这些物品在涉及多个候选分类的情况下有潜在的优

势,这将在 23.7 节进行讨论。这是因为一个分类应该对任何远离决策边界的物品都非常确定,但是如果新获取的训练数据显示分类不准确,那么该分类可能不适合该用户偏好,所以应该从候选分类池中删掉该分类。

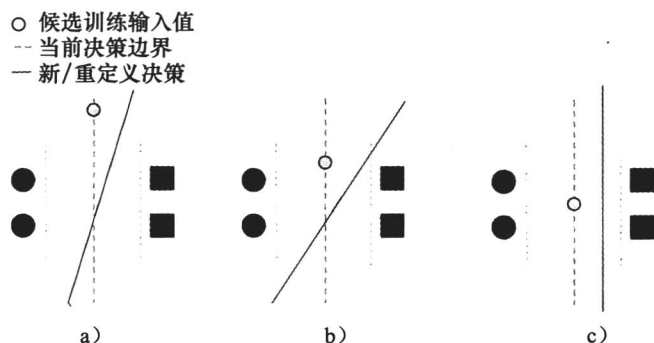


图 23.4 决策边界不确定性

### 23.5.3 模型不确定性

基于模型不确定性方法选择训练集来降低模型内的不确定性,更具体地说是为了降低模型参数的不确定性。这里假设如果提高了模型参数的准确性输出值的准确性也会随之提高。如果我们基于不同兴趣组的成员预测一个用户的偏好[23],如一组具有相似兴趣的用户,那么选择的训练集用来决定该用户属于哪一组。

#### 23.5.3.1 概率模型

下面的例子能够很好地解释概率模型。主题模型[23]是一个概率潜在语义模型,其中用户被看作多重兴趣(称为主题)的混合,该模型是一个不错的例子。对于每个  $u \in U$  在不同兴趣组  $z \in Z$  上有一个概率分布。相同兴趣组的用户被认为有相同评分形式(例如,同一视面的两个用户会对给定电影评分相同),所以对于给定隐类别变量  $z$  用户和  $x \in \mathcal{X}$  中的物品是相互独立的。用户  $u$  为物品  $x$  评分为  $y$  的概率可以通过下式来计算:

$$P(y|x, u) = \sum_{z \in Z} P(y|x, z) P(z|u) \quad (23.15)$$

第一项  $p(y|x, z)$  是分类  $z$  中的用户为物品  $x$  评分  $y$  的似然(由高斯分布近似[23]),该项不依赖目标用户,代表了特定组的模型。第二项  $p(z|u)$  是目标用户  $u$  属于类别  $z$  的概率,称为用户个性化参数(由多项分布近似[23])。用户模型  $\theta_u$  由一个或多个用户个性化参数组成,如  $\theta_u = \{\theta_{uz} = p(z|u)\} z \in Z$ 。

一个传统的主动学习方法根据能够使用户模型参数  $\theta_u$  减少多少不确定性来衡量候选训练输入点  $x_a$  是否有效(例如,用户  $u$  属于兴趣组  $z$  的不确定性):

$$\hat{G}_{\theta \text{Uncertainty}}(x_a) = \text{Uncertainty}(\theta_u) \quad (23.16)$$

$$\text{Uncertainty}(\theta_u) = - \left\langle \sum_{z \in Z} \theta_{uz} | x_a, y \log \theta_{uz} | x_a, y \right\rangle_{p(y|x_a, \theta_u)} \quad (23.17)$$

其中  $\theta_u$  表示目前用户  $u$  的估计参数,  $\theta_{uz} | x_a, y$  是通过额外的训练点  $(x_a, y)$  估计的参数。由于上述标准的目的是降低目标用户属于哪个兴趣组的不确定性,所以将用户分配到单个兴趣分类有益于训练集。这个方法可能并不是对所有模型都有效的,比如,主题模型中将用户分配给多个兴趣组更有利于用户偏好的建模[23, 25]。

根据当前估计的模型  $\theta_u$  计算分布  $p(y|x, \theta_u)$  不确定性的期望是另一个潜在的缺陷。

目前估计的模型可能与真实模型有很大差距，特别是训练集数量小而需要估计的参数数量很大的时候。因此，仅在单个估计模型上执行主动学习方法可能是有误导性的[25]。让我们通过图 23.5 中的例子进行说明。用实线表示现有的四个训练集，虚线表示测试集。根据这四个训练集，最可能的决策边界是水平线(虚线)，即使真实的决策边界是垂直线(实线)。如果只通过该估计的模型选择训练集，接下来的获取的训练集就可能是沿着评估边界，而根据正确的决策边界(垂直线)这些训练集对于调整评估决策边界(水平线)是无效的。该例子说明未考虑模型不确定性就在当前估计的模型上执行主动学习是很有误导性的，特别是估计的模型与实际模型相差很大时。一个更好的策略是利用模型分布引入模型的不确定性来选择训练输入集[25]。由于在选择训练输入集时不仅估计决策边界(如水平线)，同时其他决策边界也被考虑在内，所以可以更有效地调整该决策边界。该方法已经应用在文献[25]中的概率模型中。候选训练集的效用根据其相对于最优模型参数  $\theta_u^*$  所允许模型参数  $\theta_u$  调整的程度进行评估：

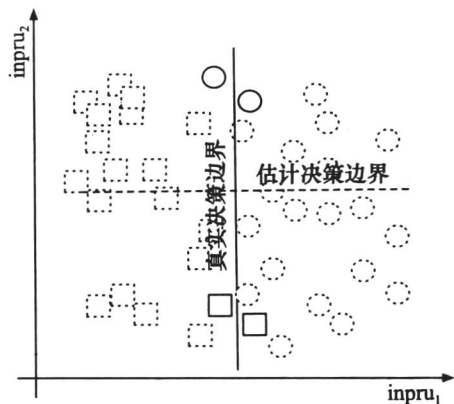


图 23.5 当评估模型与实际模型差距很大时的学习情境。训练集用实线框表示

$$\hat{G}_{\theta} |_{\text{Uncertainty}}(\mathbf{x}_a) = \left\langle \sum_{z \in Z} \theta_{u_z}^* \log \frac{\theta_{u_z} |_{\mathbf{x}_a, y}}{\theta_{u_z}^*} \right\rangle_{p(y | \mathbf{x}_a, \theta_u^*)} \quad (23.18)$$

式(23.18)相当于当  $k_l$  散度，它在估计的模型参数等于最优参数时取最小值。实际模型  $\theta_u^*$  未知，但可以通过用户模型如  $p(\theta_u | u)$  的后验分布上的期望值来估计。

## 23.6 基于误差的主动学习

基于错误的主动学习方法旨在减少预测误差，这通常也是最终目的。基于实例的方法试图找到训练输入集与预测误差之间的关系并加以利用。基于模型的方法趋向于减少模型误差(如模型参数的误差)，并期望能够因此改善预测误差。

### 23.6.1 基于实例的方法

正如在 23.2 节中所列的一样，基于实例的方法旨在输入集属性的基础上降低误差。

#### 23.6.1.1 输出估计变更(Y 变更)

文献[42]中的方法在这样一个准则下执行，即如果评分估计不变，那么方法就没有改善。因此如果输出值的估计确实改变了，那么这些输出值的正确率要么提高了要么降低了。然而，可以预计到该方法至少可以从新训练集中学到一些东西，那么可见在许多情况下估计实际上是越来越准确的。假设估计大部分是朝着有利的方向变化的，致使许多估计发生变化的物品将使得许多估计得到改善，并被认为是有益的。

举个例子(图 23.6)，如果一个用户对一个具有代表性的大类别进行评分，如科幻电影《星球大战》，那么该评分(不管值为多少)有可能会影响其他相关物品的评分估计，换句话说，为这样一个具有代表性的物品评分对于用户偏好是非常具有信息量的。另一方面，用户对没有许多相似物品的物品进行评分，如电影《金鱼猎人》，对评分估计不会带来很大影



响,同时提供的信息也很少。

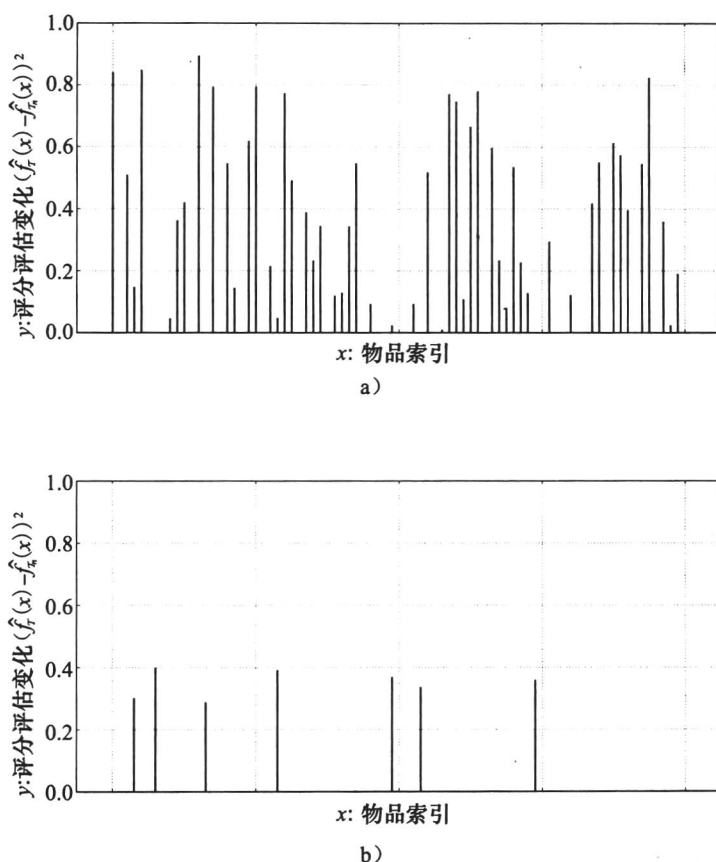


图 23.6 基于输出估计的主动学习方法(见 23.6.1.1 节)。X 轴对应物品下标, y 轴对应一个候选选训练点的评分评估变化。评分评估中变化很多的训练集被认为更有用

为找到候选物品引起的评分估计的期望变化,所有可能的物品评分都应纳入考虑(即使该物品的真正评分未知)。每个物品的每个可能评分的估计之间的差异在物品被添加到训练集之前和之后进行计算(参照算法 1 的伪代码)。

以上标准可以更正式地描述为

$$\hat{G}_{Y\text{change}}(\mathbf{x}_a) = - \sum_{\mathbf{x} \in \mathcal{X}^{(\text{Test})}} \mathbf{E}_{y \in \mathcal{Y}} \mathcal{L}(\hat{f}_\tau(\mathbf{x}), \hat{f}_{\tau U(\mathbf{x}_a, y)}(\mathbf{x})) \quad (23.19)$$

其中  $\hat{f}_\tau(\mathbf{x})$  表示在给定当前训练集  $\mathcal{T}$  时物品  $\mathbf{x}$  的评分估计,  $\hat{f}_{\tau U(\mathbf{x}_a, y)}(\mathbf{x})$  是指将物品  $\mathbf{x}_a$  的假设评分加入训练集  $\mathcal{T}$  后的评分估计,  $\mathcal{L}$  是计算评分估计  $\hat{f}_\tau(\mathbf{x})$  和  $\hat{f}_{\tau U(\mathbf{x}_a, y)}(\mathbf{x})$  差异的损失函数。通过假设一个候选物品的评分有同等机会并使用平均平方函数,上述标准可以写为

$$\hat{G}_{Y\text{change}}(\mathbf{x}_a) = - \sum_{\mathbf{x} \in \mathcal{X}^{(\text{Test})}} \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} (\hat{f}_\tau(\mathbf{x}) - \hat{f}_{\tau U(\mathbf{x}_a, y)}(\mathbf{x}))^2 \quad (23.20)$$

其中  $\frac{1}{|\mathcal{Y}|}$  是一个归一化常量,因为我们假设所有物品  $\mathbf{x}_a$  的评分为  $y \in \mathcal{Y}$ 。

该标准的优势是它只依赖于评分的估计,可以从任何学习方法中获取。还有一个区别于其他只考虑一小部分物品(被用户评过分的物品)的方法的优势,即利用所有未评分物品。该标准还配合其他任何学习方法,使得它有适应不同任务的潜力。

### 23.6.1.2 基于交叉验证的方法

在该方法中训练输入点的选择是基于训练输入点能否很好地近似已知评分,例如,文献[15]训练集中的物品。

算法 1 基于输出估计的主动学习(见 23.6.1.1 节)

---

```

#  $\hat{G}$  估计为物品  $x_a$  评分时可能会得到的预测误差
function  $\hat{G}(x_a)$ 
    # 在当前训练集  $T$  的基础上学习一个偏好近似函数
     $\hat{f}_T = \text{learn}(T)$ 
    # 对于物品  $x_a$  的每个可能的评分, 如  $\{1, 2, \dots, 5\}$ 
    对于  $y_a \in \mathcal{Y}$ 
        # 添加一个假想训练集  $(x_a, y_a)$ 
         $T^{(a)} = T \cup (x_a, y_a)$ 
        # 在新的训练集  $T^{(a)}$  上学习一个新的偏好预测函数  $\hat{f}$ 
         $\hat{f}_{T^{(a)}} = \text{learn}(T^{(a)})$ 
        # 对于每个未被评分的物品
        for  $x \in \mathcal{X}^{(\text{Test})}$ 
            # 在为训练集  $T$  添加假设训练集  $(x_a, y_a)$  之前或之后, 记录评分估计间的差异
             $\hat{G} = \hat{G} + (- (\hat{f}_T(x) - \hat{f}_{T^{(a)}}(x))^2)$ 
    return  $\hat{G}$ 

```

---

也就是, 将  $\hat{f}$  一个具有评分概率为  $y \in \mathcal{Y}$  的候选训练集  $x_a$  加入训练集  $T$ , 然后在训练集  $\mathcal{X}(\text{Train})$  上获取用户偏好并计算其准确率(如交叉验证)。假设候选训练物品包含其真正的评分时, 交叉验证的准确率将得到最大改进。候选训练集的效用通过交叉验证的准确率改进进行计算, 如下式所示:

$$\hat{G}_{CV_T}(x_a) = - \max_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}^{(\text{Train})}} \mathcal{L}(\hat{f}_{T \cup (x_a, y)}(x), f(x)) \quad (23.21)$$

其中  $\mathcal{L}$  是损失函数如 MAE 或 MSE(见 23.4 节),  $f(x)$  是物品  $x$  的真实评分,  $\hat{f}_{T \cup (x_a, y)}(x)$  是评分的近似(其中  $\hat{f}$  是从训练集  $T \cup (x_a, y)$  学习而来的)。

一个潜在的缺点是该主动学习方法选择的训练点对训练集可能过拟合。

### 23.6.2 基于模型的方法

在基于模型的方法中, 获取训练集来降低模型误差, 如模型参数的误差。这种方法的一个潜在的缺点是降低模型误差不一定会降低预测误差即主动学习的目标。

#### 23.6.2.1 基于参数变化的方法

基于参数变化的主动学习方法[49]喜欢最有可能影响模型的物品。假设这些模型参数变化是有利的, 例如, 最优参数方法中, 选择对模型参数影响最大的物品是有益的:

$$\hat{G}_{\theta \text{change}}(x_a) = - \sum_{\theta} \mathbf{E}_{y \in \mathcal{Y}} \mathcal{L}(\theta_T, \theta_{T \cup (x_a, y)}) \quad (23.22)$$

其中  $\theta_T$  是从现有训练集  $T$  中估计的模型参数,  $\theta_{T \cup (x_a, y)}$  是在物品  $x_a$  的假设评分加入训

训练集  $T$  后评估的模型参数,  $\mathcal{L}$  是计算参数之间差异的损失函数。

### 23.6.2.2 基于方差的方法

在该方法中, 误差被分解为三部分: 模型误差  $C$  (在给定当前模型上最优函数近似值  $g$  和实际函数  $f$  的差异)、偏差  $B$  (当前近似值  $\hat{f}$  与最优值  $g$  的差异) 和方差  $V$  (近似函数  $\hat{f}$  的变化大小)。换句话说有

$$G = C + B + V \quad (23.23)$$

有一个解决方法[13]是通过假设偏差部分可以忽略(如果该假设不成立那么该方法无效)来最小化误差的方差部分。有很多方法提议选择训练集来降低模型参数方差的计算。A 最优设计[11]力求选择训练输入集来最小化参数估计的平均方差, D 最优设计[26]力求最大化参数估计的香农信息差以及推导式实验(Transductive Experimental)设计[56]致力于寻找能够保留测试集大部分信息的且具有代表性的训练集。文献[51]中的主动学习方法, 除了方差部分, 还将模型误差的存在性考虑在内。

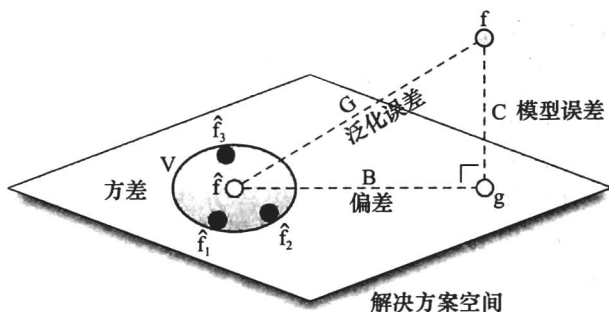


图 23.7 将泛化误差  $G$  分解为模型误差  $C$ 、偏差  $B$  和方差  $V$ , 其中  $g$  表示最优方法,  $\hat{f}$  是学习方法,  $\hat{f}_i$  是训练集有少量变化不同的学习方法

### 23.6.2.3 基于图像恢复的方法

也可以将预测用户偏好的问题看作图像恢复问题[34], 也就是根据有限的用户偏好信息(部分图片)尝试恢复完整的用户喜好图片。主动学习任务选择能够使我们恢复最好的用户偏好“图片”的训练集。值得注意的地方是该方法满足 23.2 节中概述的属性。举个例子, 如果一个区域内已经存在一个点, 那么不需要邻居集合的采样就可以恢复该区域的图片。靠近图片组成部分的边缘(决策边界)取样对该方法有利。

## 23.7 基于组合的主动学习

有时候使用模型组合而不是单一的模型可能对预测用户偏好是有益的(见第 21 章)。在只使用单个模型的情况下, 该模型是从多个候选模型中精心挑选的。这种方法的主要优势是具有这样的前提, 即不同的模型更适用于不同的用户或问题。例如, 一个用户的偏好使用原型模型建模可能会更好, 而另一个用户的偏好使用最近邻模型建模会更好。这些主动学习方法的训练集必须根据多个模型(见 23.7.1 节)或者多个候选模型(23.7.2 节)进行选择。

### 23.7.1 基于模型的方法

基于模型的方法中, 许多模型组成一个由模型形成的“组合”, 在某种意义上共同选择训练集[50]。这些方法往往在以下方面不同: 1) 如何构建一个模型组合; 2) 如何在组合成员基础上选择训练集[46]。正如文献[46]中的详细说明(请参考更详细的内容), 组合查询方法(QBC)涉及维护模型组合, 这些模型是在同样的训练数据上进行训练的。本质上, 它们表示了数据(由模型表示)可能看上去相互矛盾的假设。于是该组合的成员投票决定如何标记潜在的输入集(“QBC”中的“查询”)。为此它们最不一致的输入集被认为信息最丰富。

QBC 的基础前提是最小化版本空间或者所有假设的子集与收集的训练数据相一致, 然后尽可能地限制该空间的大小, 同时最小化训练输入集的数量。换句话说, QBC 在争议区域进行查询来改善版本空间。

构建模型组合有很多方法, 文献[46]中提供许多实例。例如, 可以通过简单的抽样进行构建[50]。对于生成模型类, 可以通过随机抽样来自某个后验分布的任意数量的模型来构建, 例如, 朴素贝叶斯模型参数上的狄利克雷分布[31], 或使用正态分布抽样隐马尔可夫模型(HMM)[14]。也可为其他模型分类构建该组合(如奇异或非概率模型), 如 boosting 查询和 bagging 查询[1], 利用 boosting[17]和 bagging[8]组合学习方法来构建组合。还有一些研究[12]为利用“最特别”与“最普通”模型相结合的(在给定的当前训练集上选择位于当前版本空间边界之间的模型)神经网络使用选择性抽样算法。

组合中所用模型的适当个数仍无定论, 但即使少量模型也可以得到好的结果[50, 31, 47]。

计算模型之间的差异是组合方法的基础, 有两种方法可以计算差异: 投票不确定性[14]和平均 KL 散度[31]。投票不确定性选择组合中模型的差异最大的点。KL 散度是两个概率分布之间差异的信息论度量方式。KL 散度从组合一致的分布与差别最大模型间的最大平均差异中选择输入集。

### 23.7.2 基于候选的方法

不同的模型会更适用于不同的用户或不同的问题(见第 2 章)。所以训练集的选择(主动学习)和模型的选择, 称为模型选择(MS), 都会影响学习到的函数的预测准确率。实际上主动学习和 MS 间有很强的依赖性, 这意味着对某个模型有用的集合对其他模型不一定有用(图 23.9)。本节讨论如何在具有多个模型候选以及过程中可能出现的问题情况下执行主动学习。

模型概念具有几个不同的含义。我们可能将一个模型看作一个具有一些共同属性的函数集, 例如, 方法的复杂度或者函数或学习方法的类型(如 SVM、朴素贝叶斯、最近邻或线性回归)。函数的不同属性一般称为参数。因此给定一个模型和训练数据, MS 的任务是找到能使目标函数精确估计的参数。所有模型属性都影响预测准确率, 但是为简单起见我们只着重于模型的复杂度。

如图 23.8 所示, 如果模型与目标函数相比太过简单使其欠拟合(图 23.8a), 那么该学习函数可能不能用来近似目标函数。另一方面, 如果模型太过复杂, 它可能会尝试近似不相关的信息(如可能包含在输出值的噪声), 这将导致学习函数过度拟合目标函数(图 23.8b)。一个可行的解决方法是包含许多候选模型。因此模型选择(MS)的目的是决定组合中模型的权重或者在只使用一个模型的情况下选择一个合适的(图 23.8c)。

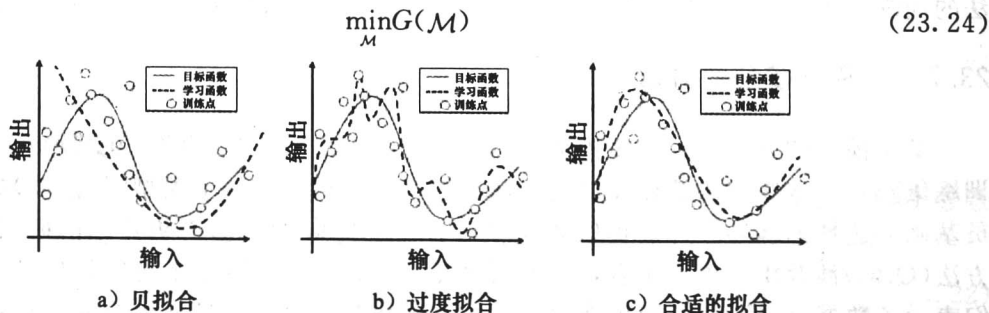


图 23.8 模型复杂度和准确率间的依赖

主动学习的任务同样是最小化预测误差，但只就训练集的选择而言：

$$\min_{\mathcal{X}^{(\text{Train})}} G(\mathcal{X}^{(\text{Train})}) \quad (23.25)$$

鉴于主动学习和 MS 有着共同的目标即最小化预测误差，把二者结合起来会是有利的：

$$\min_{\mathcal{X}^{(\text{Train})}, \mathcal{M}} G(\mathcal{X}^{(\text{Train})}, \mathcal{M}) \quad (23.26)$$

理想情况下我们会通过 MS 方法选择复杂度合适的模型以及使用主动学习方法选择最有用的训练数据。然而由于如下的悖论，只是批量地结合主动学习和 MS 方法并不可行，如一次性选择所要训练集：

- 为通过标准主动学习方法选择训练输入集，模型必须固定。换句话说，MS 方法已经执行过(见图 23.9)。

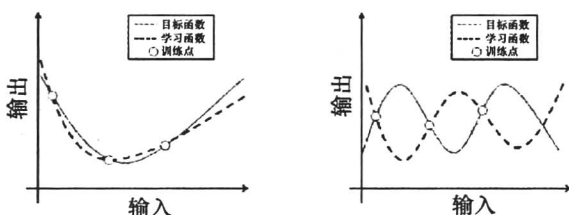


图 23.9 对一个模型有利的训练输入集不一定对其他模型有利

- 为通过标准 MS 方法选择模型，训练输入集必须固定且对应的训练输出值必须合并。换句话说，主动学习方法必须已经执行过(见图 23.10)。

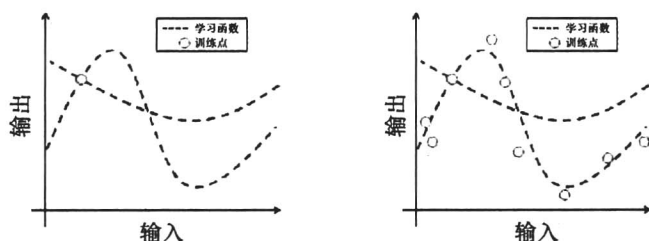


图 23.10 主动学习中的模型选择的依赖性

最终批量主动学习以一个随机选择模式来选择训练集，但如果最初和最后的模型不同，训练集可能不会那么有用，因此获取到训练集之后需要再次选择该模型。这意味着训练集可能会过度拟合或不充分拟合为劣质模型。

对于顺序主动学习，在选择模型过程中训练集和模型采取增量的过程，即选择模型然后为模型选择训练集，依次类推。虽然该方法是直观的，但有可能由于具有在整个学习过程中选择的模型不断变化导致的模型漂移使得该方法的表现会比较差。随着训练集数量的增加，更多复杂模型能够更好地拟合数据，因此选择复杂模型而不是简单模型。由于训练输入集的选择取决于模型，为学习过程前期简单模型选择的训练集对后期复杂模型可能没有用。由于模型漂移，为不同模型收集部分训练集，导致训练数据不适合其中任何模型。然而因为一开始最终模型的选择并不确定，有一种可能性是通过为所有模型优化训练数据在多个模型上[52]选择训练输入集：

$$\min_{\mathcal{X}^{(\text{Train})}} \sum_{\mathcal{M}} \hat{G}(\mathcal{X}^{(\text{Train})}, \mathcal{M}) w(\mathcal{M}) \quad (23.27)$$

其中  $w(\mathcal{M})$  作为组合或候选中模型的权重。这允许每个模型都可以优化训练数据，因此可以避免过度拟合训练集为劣质模型的可能风险。



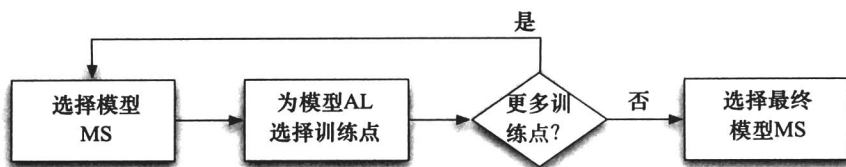


图 23.11 顺序主动学习

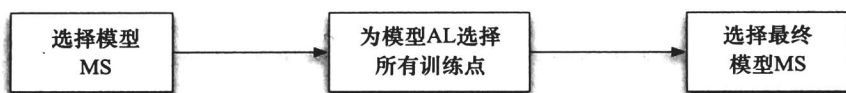


图 23.12 批量主动学习

## 23.8 基于会话的主动学习

与标准主动学习方法以获取不同物品的评分(为提高整个集合上的预测准确度)为目标不同,基于会话的主动学习是面向目标的,开始的时候都很宽泛,经过一系列的交互周期缩小用户兴趣范围直到获得所需要的物品[32, 37, 9],就像在旅行中选旅店。实质上该目标是为用户提供能够使他们减少可能的物品集合的信息来寻找最有效用的物品。因而该系统是以为物品提供正确预测为目标的,这些物品是潜在的小组物品中效用最高的,如在指定的地区查询餐馆。一个常用的方法是迭代为用户展示可变的推荐集合,并通过取得反馈来指导用户迈向最终目标,其中兴趣范围降低为单个物品。由于一开始用户几乎不知道自己的偏好(变得有主见),但在决策制定过程倾向于形成与加强偏好(探索),因此这个基于循环的方法是有用的。因此基于会话的主动学习允许用户以适合给定任务的方式加强他们的偏好。不像一般推荐系统,这种系统也特意包含主动学习方法,因为用户偏好是从主动交互中学习而来的。在达到期望目标之前,它们经常用预测准确率和交互长短来评估。

### 23.8.1 基于实例的评论

一个与用户进行会话的方法为基于实例评论的方法,该方法寻找与用户查询或属性相似的实例然后引出一个评论来改善用户兴趣(见[37]及第3章)。如上面所提到的(23.8节),在会话初始时用户并不需要明确地定义他们的偏好,这可能对面向手机设备的系统特别有益。每一步迭代显示系统推荐的排名列表并允许用户评论,这将促使系统重新评估它的推荐并产生一个新的排名列表。当不满意某个被推荐物品的特征时,通过引出一个用户评论来获取最终目标要比通过单纯的基于相似性查询修改结合推荐建议更有效。举一个用户评论的例子,他/她可能会添加评论“我想要一个较便宜的客房”或“我喜欢提供葡萄酒的旅馆”。

### 23.8.2 基于多样性的方法

为用户提供与其查询相似的物品是重要的(23.8.1节),同时推荐物品集合的多样性也值得考虑[32]。这是因为被推荐的物品互相太过相似,它们可能不是目前的搜索空间中

具有代表性的。实际上,被推荐的物品应尽可能地具有代表性和多样性,而且不会明显地影响它们与用户查询的相似性。

在用户偏好形成的初始阶段为其提供多样的选择尤其重要。一旦用户知道他们想要什么,提供尽可能匹配的物品是合适的,并且使用的主动学习技术应该尝试进行这种区分,例如,如果推荐空间被适当集中,则减少多样性,如果不正确,则增加多样性。

### 23.8.3 基于查询编辑的方法

另一种方法是允许用户重复编辑和提交搜索,直到他们发现期望的物品[9]。由于它是一个迭代的过程,其目标是在用户找到效用最高的物品之前最小化所需查询数量。查询的效用在用户提交的特定查询的相似度和满意度基础上来评估,通过观察用户行为和推测与物品效用有关的用户偏好上的约束及更新用户模型来完成。例如,一个用户可能会查询带有空调和高尔夫课程的旅馆。推荐系统能够确定这个条件是可以满足的,并进一步推断该用户可能会加一个旅馆在市中心的条件,而没有旅馆符合要求,所以系统先发制人告诉该用户该条件无法满足以防浪费用户精力。推荐系统也可能推断有些旅馆自带泳池、SPA和餐馆,只需稍微增加一下价钱。知道用户对泳池的偏好(对其他选项没有),系统只提供添加泳池选项,因为这可能会增加用户的满意度,而且系统不提供其他选项,因为可能会让用户感觉冗余从而降低整体满意度。

## 23.9 计算因素考虑

考虑主动学习算法的计算代价也是很重要的。文献[40]中提出很多降低计算需求的方法,总结(并做一些补充)如下:

- 许多主动学习方法在其对学习到的函数的期望效果基础上选择物品。这可能需要重新训练每个候选训练物品,所以有效地增量训练是至关重要的。通常比从一个大的集合开始这种循序渐进的方法的成本更低。
- 对每个候选物品可能都需要重新获取评分估计。同样也使用增量方法,因为只有变化的估计才需要重新获取。
- 只对受包含训练点影响的物品以增量方式更新估计错误是可行的,实际上这些只是训练点附近的物品或者没有相似特征的物品。一个常用的方法是快速查询使用倒排索引对具有相似特征的物品进行分组。
- 候选训练物品的期望效用可以使用所有物品的子集进行评估。
- 训练集中的劣质候选可以通过一个预过滤步骤进行删减,该步骤根据某些标准移除劣质候选物品,例如,过滤掉使用用户无法阅读的语言书写的书。对于该任务次优主动学习方法可能是一个不错的选择。

## 23.10 总结

虽然本章所展示的主动学习方法集合的介绍都很简单,但希望已经阐述明白对许多系统来说,即推荐系统,主动学习不仅是有利的也是理想的方法。可以看出,由于个体特征,或由于业务约束、首选系统行为、用户经验或以上条件的组合(也有可能是其他的条件),在许多情况下主动学习方法的选择很大程度上取决于必须满足的具体目标(见 23.1.1

节)。除了主动学习的目标,在评估任何方法的使用和权衡计算代价(见 23.9 节)时也需要谨慎。尽管许多讨论过的方法都已成功实现,但还需要简单说一下存在的问题,或为其他看似无关但可能有相似解决办法的问题寻找解决方法(如 23.6.2.3 节中的图像恢复方法)。我们还涉及基于会话的系统(见 23.8 节),该系统与传统推荐系统不同,但从设计上就包含主动学习概念。根据现有任务,如面向具体目标的辅助系统,该系统可能是推荐系统的不错选择。

一些与主动学习相关的问题在统计学中已经得到充分的研究。而计算机科学领域情况不同,还需要进行研究。推荐系统变化非常快而且越来越复杂。赢得 Netflix 推荐竞赛是一个关于推荐系统的例子,其中多种预测方法通过组合方法集合起来(见第 5 章)。鉴于推荐系统中预测方法的高变化率以及主动学习的复杂交互,越来越需要新的方法。

提高准确率通常是研究的着重之处。而仅仅有准确率并不能增加系统的用户黏性。因为系统在完成主动学习方法时还需要推荐具有高新颖性/惊喜度的物品、提高覆盖率或最大化利润等[44, 21, 33]。经常被主动学习研究者忽略的另一个方面是可以使用户与主动学习的交互来改进性能的方法。简单地展示物品让用户评分至少可以说是缺乏创造性的,那么一定有更好的方法吗?有这样一个例子,文献[3]中表明使用正确的用户界面甚至像标记图片这样的小事可以是有趣并令人兴奋的。只是使用正确的界面主动学习系统的效用就有可能急剧增加。

为保证主动学习在推荐系统中的长期使用,许多遗留问题必须要解决。通过点滴的创新和艰苦的工作,我们希望看到主动学习从一个“繁重的过程”变为一个愉快的自我发现和探索的过程,并同时满足系统目标和用户。

## 致谢

非常感谢 Okamoto 教授、Ueno 教授、Tokunaga 教授、Tomioka 教授、Sheinman 博士、Vilenius 博士、Sachi Kabasawa 和 Akane Odake 的帮助和协助,也要感谢 MEXT 和 JST 的赞助。审稿人和编辑的评论也是写作过程中必不可少的一环。

## 参考文献

1. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging. In: Proceedings of the Fifteenth International Conference on Machine Learning, vol. 388. Morgan Kaufmann Publishers Inc. (1998)
2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
3. Ahn, L.V.: Games with a purpose. *Computer* 39(6), 92–94 (2006). DOI 10.1109/MC.2006.196
4. Bailey, R.A.: Design of Comparative Experiments. Cambridge University Press (2008)
5. Balcan, M.F., Beygelzimer, A., Langford, J.: Agnostic active learning. In: ICML '06: Proceedings of the 23rd international conference on Machine learning, pp. 65–72. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1143844.1143853>
6. Boutilier, C., Zemel, R., Marlin, B.: Active collaborative filtering. In: Proceedings of the Nineteenth Annual Conference on Uncertainty in Artificial Intelligence, pp. 98–106 (2003). URL [citeseer.ist.psu.edu/boutilier03active.html](http://citeseer.ist.psu.edu/boutilier03active.html)
7. Box, G., Hunter, S.J., Hunter, W.G.: Statistics for Experimenters: Design, Innovation, and Discovery. Wiley-Interscience (2005)

8. Breiman, L., Breiman, L.: Bagging predictors. In: *Machine Learning*, pp. 123–140 (1996)
9. Bridge, D., Ricci, F.: Supporting product selection with query editing recommendations. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 65–72. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297243>
10. Carenini, G., Smith, J., Poole, D.: Towards more conversational and collaborative recommender systems. In: *IUI '03: Proceedings of the 8th international conference on Intelligent user interfaces*, pp. 12–18. ACM, New York, NY, USA (2003). DOI <http://doi.acm.org/10.1145/604045.604052>
11. Chan, N.: A-optimality for regression designs. Tech. rep., Stanford University, Department of Statistics (1981)
12. Cohn, D.A.: Neural network exploration using optimal experiment design 6, 679–686 (1994). URL [citeseer.ist.psu.edu/article/cohn94neural.html](http://citeseer.ist.psu.edu/article/cohn94neural.html)
13. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. *Journal of Artificial Intelligence Research* 4, 129–145 (1996)
14. Dagan, I., Engelson, S.: Committee-based sampling for training probabilistic classifiers. In: *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 150–157. Citeseer (1995)
15. Danziger, S., Zeng, J., Wang, Y., Brachmann, R., Lathrop, R.: Choosing where to look next in a mutation sequence space: Active learning of informative p53 cancer rescue mutants. *Bioinformatics* 23(13), 104–114 (2007)
16. Dasgupta, S., Lee, W., Long, P.: A theoretical analysis of query selection for collaborative filtering. *Machine Learning* 51, 283–298 (2003). URL [citeseer.ist.psu.edu/dasgupta02theoretical.html](http://citeseer.ist.psu.edu/dasgupta02theoretical.html)
17. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences* 55(1), 119–139 (1997)
18. Fujii, A., Tokunaga, T., Inui, K., Tanaka, H.: Selective sampling for example-based word sense disambiguation. *Computational Linguistics* 24, 24–4 (1998)
19. Greiner, R., Grove, A., Roth, D.: Learning cost-sensitive active classifiers. *Artificial Intelligence* 139, 137–174 (2002)
20. Harpale, A.S., Yang, Y.: Personalized active learning for collaborative filtering. In: *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 91–98. ACM, New York, NY, USA (2008). DOI <http://doi.acm.org/10.1145/1390334.1390352>
21. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* 22(1), 5–53 (2004). DOI <http://doi.acm.org/10.1145/963770.963772>
22. Hinkelmann, K., Kempthorne, O.: *Design and Analysis of Experiments*, Advanced Experimental Design. Wiley Series in Probability and Statistics (2005)
23. Hofmann, T.: Collaborative filtering via gaussian probabilistic latent semantic analysis. In: *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 259–266. ACM, New York, NY, USA (2003). DOI <http://doi.acm.org/10.1145/860435.860483>
24. Huang, Z.: Selectively acquiring ratings for product recommendation. In: *ICEC '07: Proceedings of the ninth international conference on Electronic commerce*, pp. 379–388. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1282100.1282171>
25. Jin, R., Si, L.: A bayesian approach toward active learning for collaborative filtering. In: *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pp. 278–285. AUAI Press, Arlington, Virginia, United States (2004)
26. John, R.C.S., Draper, N.R.: D-optimality for regression designs: A review. *Technometrics* 17(1), 15–23 (1975)
27. Kapoor, A., Horvitz, E., Basu, S.: Selective supervision: Guiding supervised learning with decision-theoretic active learning. In: *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 877–882 (2007)
28. Kohrs, A., Merialdo, B.: Improving collaborative filtering for new users by smart object selection. In: *Proceedings of International Conference on Media Features (ICMF)* (2001)
29. Leino, J., Räihä, K.J.: Case amazon: ratings and reviews as part of recommendations. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 137–140. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297255>
30. Lomasky, R., Brodley, C., Aerncke, M., Walt, D., Friedl, M.: Active class selection. In: *Proceedings of the European Conference on Machine Learning (ECML)*. Springer (2007)
31. McCallum, A., Nigam, K.: Employing em and pool-based active learning for text classification.

- cation. In: ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning, pp. 350–358. San Francisco, CA, USA (1998)
32. McGinty, L., Smyth, B.: On the Role of Diversity in Conversational Recommender Systems. *Case-Based Reasoning Research and Development* pp. 276–290 (2003)
33. McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: CHI '06: CHI '06 extended abstracts on Human factors in computing systems, pp. 1097–1101. ACM Press, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1125451.1125659>
34. Nakamura, A., Abe, N.: Collaborative filtering using weighted majority prediction algorithms. In: ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning, pp. 395–403. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
35. Rashid, A.M., Albert, I., Cosley, D., Lam, S.K., McNee, S.M., Konstan, J.A., Riedl, J.: Getting to know you: learning new user preferences in recommender systems. In: IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces, pp. 127–134. ACM Press, New York, NY, USA (2002). DOI <http://doi.acm.org/10.1145/502716.502737>
36. Rashid, A.M., Karypis, G., Riedl, J.: Influence in ratings-based recommender systems: An algorithm-independent approach. In: SIAM International Conference on Data Mining, pp. 556–560 (2005)
37. Ricci, F., Nguyen, Q.N.: Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems* 22(3), 22–29 (2007). DOI <http://dx.doi.org/10.1109/MIS.2007.43>
38. Rokach, L., Naamani, L., Shmilovici, A.: Pessimistic cost-sensitive active learning of decision trees for profit maximizing targeting campaigns. *Data Mining and Knowledge Discovery* 17(2), 283–316 (2008). DOI <http://dx.doi.org/10.1007/s10618-008-0105-2>
39. Rokach, L. and Maimon, O. and Arbel, R., Selective voting-getting more for less in sensor fusion, *International Journal of Pattern Recognition and Artificial Intelligence* 20 (3) (2006), pp. 329–350.
40. Roy, N., McCallum, A.: Toward optimal active learning through sampling estimation of error reduction. In: In Proc. 18th International Conf. on Machine Learning, pp. 441–448. Morgan Kaufmann (2001)
41. Rubens, N., Sugiyama, M.: Influence-based collaborative active learning. In: Proceedings of the 2007 ACM conference on Recommender systems (RecSys 2007). ACM (2007). DOI <http://doi.acm.org/10.1145/1297231.1297257>
42. Rubens, N., Tomioka, R., Sugiyama, M.: Output divergence criterion for active learning in collaborative settings. *IPSI Transactions on Mathematical Modeling and Its Applications* 2(3), 87–96 (2009)
43. Saar-Tsechansky, M., Provost, F.: Decision-centric active learning of binary-outcome models. *Information Systems Research* 18(1), 4–22 (2007). DOI <http://dx.doi.org/10.1287/isre.1070.0111>
44. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 253–260. ACM, New York, NY, USA (2002). DOI <http://doi.acm.org/10.1145/564376.564421>
45. Schohn, G., Cohn, D.: Less is more: Active learning with support vector machines. In: Proc. 17th International Conf. on Machine Learning, pp. 839–846. Morgan Kaufmann, San Francisco, CA (2000). URL [citeseer.ist.psu.edu/schohn00less.html](http://citeseer.ist.psu.edu/schohn00less.html)
46. Settles, B.: Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison (2009)
47. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1069–1078. ACL Press (2008)
48. Settles, B., Craven, M., Friedland, L.: Active learning with real annotation costs. In: Proceedings of the NIPS Workshop on Cost-Sensitive Learning, pp. 1–10 (2008)
49. Settles, B., Craven, M., Ray, S.: Multiple-instance active learning. In: Advances in Neural Information Processing Systems (NIPS), vol. 20, pp. 1289–1296. MIT Press (2008)
50. Seung, H.S., Opper, M., Sompolsky, H.: Query by committee. In: Computational Learning Theory, pp. 287–294 (1992). URL [citeseer.ist.psu.edu/seung92query.html](http://citeseer.ist.psu.edu/seung92query.html)
51. Sugiyama, M.: Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research* 7, 141–166 (2006)
52. Sugiyama, M., Rubens, N.: A batch ensemble approach to active learning with model selection. *Neural Netw.* 21(9), 1278–1286 (2008). DOI <http://dx.doi.org/10.1016/j.neunet.2008.06.004>



53. Sugiyama, M., Rubens, N., Mueller, K.R.: Dataset Shift in Machine Learning, chap. A conditional expectation approach to model selection and active learning under covariate shift. MIT Press, Cambridge (2008)
54. Swearingen, K., Sinha, R.: Beyond algorithms: An hci perspective on recommender systems. ACM SIGIR 2001 Workshop on Recommender Systems (2001).
55. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In: P. Langley (ed.) Proceedings of ICML-00, 17th International Conference on Machine Learning, pp. 999–1006. Morgan Kaufmann Publishers, San Francisco, US, Stanford, US (2000). URL [citeseer.ist.psu.edu/article/tong01support.html](http://citeseer.ist.psu.edu/article/tong01support.html)
56. Yu, K., Bi, J., Tresp, V.: Active learning via transductive experimental design. In: Proceedings of the 23rd Int. Conference on Machine Learning ICML '06, pp. 1081–1088. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1143844.1143980>

## 多准则推荐系统

Gediminas Adomavicius、Nikos Manouselis 和 YoungOk Kwon

**摘要** 本章主要目标是对多准则推荐系统进行综述。首先，本章将推荐问题定义为多准则决策问题(Mutli-criteria Decision Making, MCDM)，并回顾了用来支撑多准则推荐系统实现的 MCDM 方法及技术。接着本章将围绕一类称为多准则评分推荐系统(multi-criteria rating recommenders)的推荐技术，该推荐技术通过多个准则的评分向量来建模用户对每个项目的效用，进而产生推荐。文中总结了已有的利用多准则评分进行预测和推荐的算法。最后，还对多准则推荐系统的一些尚未解决的问题以及未来面临的挑战进行讨论。

### 24.1 简介

推荐问题被定义为一种用来帮助社区中的个体用户发现那些他们最有可能感兴趣或者与他们的需求相关的信息和物品[4, 39, 72]。通常，假设系统的用户集为 Users，可推荐的候选物品集为 Items，效用函数用于度量推荐物品  $i(i \in \text{Items})$  与用户  $u(u \in \text{Users})$  之间的适合度，定义为： $R: \text{Users} \times \text{Items} \rightarrow R_0$ ，其中  $R_0$  表示在一定范围内的非负整数或实数[4]。假定此效用函数不是在所有  $\text{Users} \times \text{Items}$  空间里都是已知的，而是仅在一定的子集范围内确定的。因此，在推荐上下文中，我们希望每个用户  $u(u \in \text{Users})$  都能够：①对于物品  $i(i \in \text{Items})$  未知的效用函数  $R(u, i)$  进行估计；②选择一个或一组物品  $i(i \in \text{Items})$ ，使得  $R(u, i)$  达到最大值，即

$$\forall u \in \text{Users}, i = \arg \max_{i \in \text{Items}} R(u, i) \quad (24.1)$$

在绝大多数的推荐系统中，效用函数通常认为是一个单准则值，例如，一个用户对一个物品的整体性评估或评分值。最近的研究表明，这一假设存在一定的局限性[2, 4, 48]，因为对于一个特定的用户而言，推荐物品的合适度可能不仅只依赖于用户做决策时的某一个方面考虑。尤其是在基于其他用户观点的推荐系统中，多重准则的引入可以影响用户自身的观点，从而产生更准确的推荐。

因此，多准则评分(multi-criteria rating)提供的额外信息可以改善推荐质量，这是由于它能够反映每个用户更复杂的偏好信息。下面是关于多准则评分的案例说明，在传统的单评分电影推荐系统中，用户  $u$  对他看过的电影  $i$  给出了一个评分，记作  $R(u, i)$ 。具体地，假设推荐系统是依据那些具有相似偏好信息的用户对这部

Gediminas Adomavicius, YoungOk Kwon, Department of Information and Decision Sciences Carlson School of Management, University of Minnesota, Minneapolis, MN 55455, USA e-mail: {gedas, kwonx052}@umn.edu

Nikos Manouselis, Greek Research and Technology Network (GRNET S.A.) 56 Messogeion Av., 115 27, Athens, Greece e-mail: nikosm@grnet.gr

翻译：张秀伟 审核：朱亮，李青，李曙光

电影的评分来预测用户对没有观看过的电影的评分,通常这些具有相似偏好信息用户被称作“邻居”[71]。因此为了获得准确的预测或推荐,准确确定与目标用户最相似的用户群的能力对产生精确预测和推荐至关重要。例如,两个用户  $u$  和  $u'$  观看了三部相同的电影,两人对每一部电影的综合满意度评分都为 6 分(满分 10 分),于是这两个用户就可以认为是邻居用户,用户  $u$  对未观看电影的评分可以依据用户  $u'$  的评分来预测。

相比之下,在多准则评分环境中,用户可以对一个物品的多个属性进行评分。例如,在一个双准则电影推荐系统中,用户可以对电影的两个属性(例如,故事情节和视觉效果)给出自己的偏好。某一个用户可能喜欢一部电影的故事情节,而不喜欢它的视觉效果,例如,  $R(u, i) = (9, 3)$ 。如果我们简单地使用两个相同权值的评分来进行推荐,那么对于在单一评分系统中综合满意度为 6 分(满分 10 分)的情况,在双准则系统中可能会对多种情况:  $(9, 3)$ ,  $(6, 6)$ ,  $(4, 8)$  等。因此,尽管整体满意度都是 6,但是两个用户可能采用完全不同的评分模式,例如,对于同样的三部电影,用户  $u$  的评分是  $(9, 3)$ ,  $(9, 3)$ ,  $(9, 3)$ , 而用户  $u'$  的评分是  $(3, 9)$ ,  $(3, 9)$ ,  $(3, 9)$ 。用户偏好中的额外信息将有助于更准确地对用户的偏好建模,需要开发新的推荐技术来利用这些额外信息。由于多准则推荐系统的重要研究价值,已经在推荐系统的相关文献中凸显出来成为一个研究分支[2, 4, 48]。在后续章节中我们会讲到,最近也有一些推荐系统采用了多准则评分来替代传统的单准则评分。所以本章主要目的是提供一个**多准则推荐系统**(multi-criteria recommender system)的综述,特别是**多准则评分**(multi-criteria rating)方面。

本章的后续部分组织安排如下:首先,综述了在多准则决策(MCDM)视角下的一般性推荐问题,并论证了在多准则环境中应用 MCDM 方法提升推荐能力的可行性;其次,我们关注于采用多准则评分这一特定类型的多准则推荐系统,称为多准则评分推荐系统,尽管这类系统还没有被广泛研究,但是它对于推荐性能的改善还是具有很大的潜力的。我们对这一类推荐系统的算法现状进行综述。最后,我们讨论了多准则推荐系统领域的研究挑战以及未来的研究方向。

## 24.2 推荐作为多准则决策问题

为了将多准则引入一般的推荐问题,可遵循 MCDM 的经典方法之一。为了便于探讨 MCDM 方法和技术在开发推荐系统中的应用,我们在决策支持中的通用建模方法中,沿用了 Bernard Roy(20 世纪 60 年代 MCDM 研究的先驱者)提出的步骤和符号[77]。这一讨论同样遵循了其他的 MCDM 建模方法[24, 34, 95, 97]。本节范围是为了给推荐系统研究人员在考虑设计一个多准则推荐系统时,提供一些初步的指导。

Roy[77]的方法在分析决策支持过程中包括四个步骤:

- **定义决策目标。**即定义制定决策所依赖的候选物品集和推荐决策的理论依据。
- **定义一致的准则簇。**即标识和指定一个用来说明决策者面对多种选择的偏好的函数集。这些函数集需要涵盖所有影响推荐决策的参量,并且是完整和无冗余的。
- **开发一个全局性的偏好模型。**定义一个用来将每个准则的偏好集成到模型中函数,确定决策者针对一个候选物品整体偏好。
- **选择决策支持流程。**确切地说,包括对流程、方法或者软件系统的设计和开发,以用来支持决策者对不同情况做出决策。

我们在 24.2.1 节中简述上述步骤是如何适用于推荐系统的。

### 24.2.1 决策目标

在推荐系统中,决策目标就是将物品  $i$  放到所有候选物品的哪个集合中,这个集合的元素在相关文献中称为可选物品或者动作[30]。为了表达决策背后的理论依据,Roy[77]引入了决策“问题”(problematic)的概念。定义了四类决策问题:

- 选择(choice),关注于在候选集中选择一个或多个可选项作为更合适的选择;
- 分类(sorting),指的是将可选项分类到多个预先定义类别中;
- 排序(ranking),指的是对所有的可选项由好到坏进行排序;
- 描述(discription),即如何描述它与准则之间的符合度。

所有四类决策问题在推荐问题中都是适用的:

- 为特定的用户选择并推荐一个或多个更合适的物品;
- 分类,根据物品的适用性将所有的物品分到预先定义类别中,比如,按照“推荐购买物品”或“推荐浏览物品”;
- 针对特定的用户,将所有的物品从最合适到最不合适进行排序,然后将排好序后的推荐结果列表展现给用户;
- 基于每个准则评估,描述特定物品是如何适合于指定用户的。也相当于一个对物品性能在所有准则上的全面分析,描述一个物品和用户之间的合适度(即用个性化方法帮助用户做出选择)。

### 24.2.2 准则簇

物品集中的可选项的效用是依据每个用户的准则集分析得来的,以便于对物品所有的特征、属性、效果或因果关系建模[77, 97]。在推荐系统中,准则可能涉及物品的多个特征(通常是在基于内容的推荐中),或涉及被评估或评分的物品的多个维度。

任何一个准则  $c$  都可以用函数  $g_c(i)$  用来表达一个用户的偏好(由用户自定义),以便于用户能够在两个可选项  $i_1$  和  $i_2$  之间做出决定,若  $g_c(i_1) > g_c(i_2)$ ,说明  $i_1$  比  $i_2$  更适合用户,若  $g_c(i_1) = g_c(i_2)$  则表示这两个物品是等价的(即在这个准则下可以为用户完美替换的选项)。为了能在多准则中做出合理的抉择,需要确保整个函数集创建了一致性的准则簇[77]。一致性的准则簇必须满足以下三个特性:

- 单调性(monotonic):准则簇的单调性是指对于每一对可选项  $i_1$  和  $i_2$ ,如果在准则  $c_1$  上,有  $g_{c_1}(i_1) > g_{c_1}(i_2)$ ;在任何其他准则  $c$  上,有  $g_c(i_1) = g_c(i_2)$ ,  $c \neq c_1$ ;则认为,可选项  $i_1$  比  $i_2$  更适合用户。
- 完整性(exhaustive):准则簇的完整性是指对于每一对可选项  $i_1$  和  $i_2$ ,如果在每一个准则  $c$  上,有  $g_c(i_1) = g_c(i_2)$ ;则认为,  $i_1$  和  $i_2$  是等价的。
- 非冗余性(non-redundant):准则簇的非冗余性是指如果移除任意一个准则,都会违背其他两条特性中的一条。

在本章的剩余部分,除非特别说明,假定准则簇  $k$  为一致性的准则簇,即  $g_1, g_2, g_3, \dots, g_k$ 。在现有的一些推荐系统文献里,准则簇的一致性的设计在很大程度上都被忽略了,其实它将是未来研究中的一个非常有趣和重要的问题。在 MCDM 中通常有四种类

型的准则[30]:

- **可度量的(measurable)**, 即在某些评价标准上可以对准则进行定量的度量;
- **序数化的(ordinal)**, 即准则定义的有序集合中可以接受的值, 允许进行定性或描述性的评价;
- **概率化的(probabilistic)**, 即在对准则的评价中使用概率分布表示不确定性;
- **模糊性的(fuzzy)**, 即准则的量化是用来表示它所使用的定性的或定量的量化尺度之间的可能性关系准则的评价可以看作落在定性或描述性评价的某个区间概率的关系。

广义上来讲, 准则簇通常也可以用于推荐系统, 以便更好地表达用户的偏好。因此, 我们可以假定所有类型的准则都可以用于多准则推荐系统, 尽管有些类型(稍后说明)比其他类型在当前成熟系统中使用得更频繁。

### 24.2.3 全局偏好模型

全局偏好模型提供了一种把每个准则值  $g_c (c=1, \dots, k)$  进行聚合的方法, 用来表达在物品集中不同选项之间的偏好, 它取决于选择问题类型。在 MCDM 文献中, 提出了多种方法, 这些方法可以依照全局偏好模型的使用以及创建方式被分成不同类别。根据文献[30, 64], 全局偏好模型可以分为以下几类:

- **价值导向模型(Value-Focused model)**, 它构造了由不同准则的用户偏好聚合而产生的价值体系。在这种方法中, 每个准则上的边界偏好被集成为一个总价值函数, 这一函数通常称为效用函数(Utility Function)[33], 这些方法又称为多属性效用理论方法(MAUT)。
- **多目标优化模型(Multi-Objective Optimization model)**, 它的准则使用了多约束条件对多目标优化问题的这种表达形式。该方法中, 目标通常是为了原始优化问题去寻找帕雷托(Pareto)最优解决方案[102]。该类方法也被称成为多目标数学规划方法。
- **级别优先关系模型(Outranking Relations model)**, 在该模型中, 偏好用来表示物品之间的级别优先关系, 因此允许它们有不可比性。在这类方法中, 所有的物品都是成对元素之间的互相比, 并且它们之间的偏好关系用关系名词来表示为, 如“ $a$  比  $b$  更受偏爱”, “ $a$  与  $b$  同样受偏爱”, 或“ $a$  与  $b$  无法相比”[76]。
- **偏好分解模型(Preference Disaggregation model)**, 它的偏好模型是经过对以往决策的分析推导得出的。这类方法有时候作为以上模型的子类, 因为他们需要依照过去导致独特决策的一些偏好结构来推测一个指定形式(如价值函数, 或者级别优先关系)的偏好模型。隐含偏好模型的目的是在于使产生的决策过程中与以往决策要至少保持一致[30]。

所有种类中的方法都可以为推荐系统创建全局偏好模型, 这依赖于选定的决策问题类型和推荐系统预期的使用环境。

### 24.2.4 决策支持流程

在这一步中, 一个给定的 MCDM 问题的最终决策是通过选择一个合适方法来做决策, 而这个方法又源于之前的每一步所定义的方法之中。与传统的 MCDM 问题



一样，多准则推荐问题在不同的领域和应用中采用不同的方法。需要注意的是，在大多准则推荐问题建模时，所谓的 MCDM 概念很宽泛，并没有严格的界限，这是因为许多现有的推荐系统都可以归入 MCDM 中的这一类中，由于它们通常考虑多源信息（如用户概况和物品属性），事实上就将它变成了实际上的多准则决策者。因此，在本章的后续部分，我们关注于 MCDM 推荐系统中的特殊一类，它们与现有的推荐系统大不相同。

在表 24.1~表 24.3 里，我们基于文献[48]所述，列举了一些推荐系统，这些推荐系统可以大致归类为 MCDM(或多准则推荐)。这个综述所涉及的推荐系统都使用了前面章节里提到过的 MCDM 方法之一，所以，通过对现有的 MCDM 方法的了解，可以使它们应用到推荐系统中辅助决策。

表 24.1 已有多准则推荐系统支持的决策问题

选择	Ariely et al. 2004 [6], Falle et al. 2004 [23], Kleinberg and Sandler 2003 [38], Lee et al. 2002 [45], Lee 2004 [44], Price and Messinger 2005 [69], Tewari et al. 2003 [93]
分类	Cantador et al. 2006 [12], Choi and Cho 2004 [15], Emi et al. 2003 [22], Guan et al. 2002 [28], Kim and Yang 2004 [36], Liu and Shih 2005 [47], Masthoff 2003 [53], Montaner et al. 2002 [57], Nguyen and Haddawy 1999 [60], Nguyen and Haddawy 1998 [59], Stolze and Rjaibi 2001 [90], Wang 2004 [99], Yu 2002 [100], Yu 2004 [101], Zimmerman et al. 2004 [103]
排序	Adomavicius and Kwon 2007 [2], Ardissono et al. 2003 [5], Balabanovic and Shoham 1997 [7], Ghosh et al. 1999 [27], Karacapilidis and Hatzieleftheriou 2005 [32], Kerschberg et al. 2001 [35], Kim et al. 2002 [37], Lakiotaki et al. 2008 [42], Lee and Tang 2007 [43], Lee et al. 2002 [45], Li et al. 2008 [46], Manouselis and Costopoulou 2007b [49], Manouselis and Costopoulou 2007c [50], Manouselis and Sampson 2004 [52], Mukherjee et al. 2001 [58], Noh 2004 [61], Perny and Zucker 1999 [66], Perny and Zucker 2001 [67], Plantie et al. 2005 [68], Ricci and Werthner 2002 [73], Ricci and Nguyen 2007 [74], Sahoo et al. 2006 [79], Schafer 2005 [82], Schickel-Zuber and Faltings 2005 [83], Srikumar and Bhasker 2004 [89], Tang and McCalla 2009 [92], Tsai et al. 2006 [96]
描述	Aciar et al. 2007 [1], Cheetham 2003 [14], Denguir-Rekik et al. 2006 [19], Herrera-Viedma et al. 2004 [29], Schmitt et al. 2002 [84], Schmitt et al. 2003 [85], Stolze and Stroebel 2003 [91]

表 24.2 已有多准则推荐系统采用的准则类型

可度量	Adomavicius and Kwon 2007 [2], Ariely et al. 2004 [6], Balabanovic and Shoham 1997 [7], Cantador et al. 2006 [12], Choi and Cho 2004 [15], Falle et al. 2004 [23], Ghosh et al. 1999 [27], Guan et al. 2002 [28], Kerschberg et al. 2001 [35], Kim and Yang 2004 [36], Kim et al. 2002 [37], Lakiotaki et al. 2008 [42], Lee and Tang 2007 [43], Lee 2004 [44], Lee et al. 2002 [45], Li et al. 2008 [46], Liu and Shih 2005 [47], Manouselis and Costopoulou 2007b [49], Manouselis and Costopoulou 2007c [50], Manouselis and Sampson 2004 [52], Masthoff 2003 [53], Montaner et al. 2002 [57], Mukherjee et al. 2001 [58], Noh 2004 [61], Plantie et al. 2005 [68], Ricci and Werthner 2002 [73], Ricci and Nguyen 2007 [74], Sahoo et al. 2006 [79], Schafer 2005 [82], Schickel-Zuber and Faltings 2005 [83], Schmitt et al. 2003 [85], Schmitt et al. 2002 [84], Srikumar and Bhasker 2004 [89], Stolze and Rjaibi 2001 [90], Tang and McCalla 2009 [92], Tewari et al. 2003 [93], Tsai et al. 2006 [96], Yu 2002 [100], Yu 2004 [101], Zimmerman et al. 2004 [103]
有序	Aciar et al. 2007 [1], Cheetham 2003 [14], Emi et al. 2003 [22], Nguyen and Haddawy 1998 [59], Nguyen and Haddawy 1999 [60]
模糊	Herrera-Viedma et al. 2004 [29], Karacapilidis and Hatzieleftheriou 2005 [32], Perny and Zucker 1999 [66], Perny and Zucker 2001 [67], Stolze and Stroebel 2003 [91], Wang 2004 [99]
概率	Ardissono et al. 2003 [5], Kleinberg and Sandler 2003 [38], Price and Messinger 2005 [69]

表 24.3 已有多准则推荐系统用到的全局偏好

价值导向模型	Aciar et al. 2007 [1], Adomavicius and Kwon 2007 [2], Ariely et al. 2004 [6], Balabanovic and Shoham 1997 [7], Cantador et al. 2006 [12], Choi and Cho 2004 [15], Denguir-Rekik et al. 2006 [19], Falle et al. 2004 [23], Ghosh et al. 1999 [27], Guan et al. 200 [28], Herrera-Viedma et al. 2004[29], Karacapilidis and Hatzieleftheriou 2005 [32], Kerschberg et al. 2001 [35], Kim and Yang 2004 [36], Kim et al. 2002 [37], Kleinberg and Sandler 2003 [38], Lakiotaki et al. 2008 [42], Lee 2004 [44], Lee et al. 2002 [45], Li et al. 2008 [46], Liu and Shih 2005 [47], Manouselis and Costopoulou 2007b [49], Manouselis and Costopoulou 2007c [50], Manouselis and Sampson 2004 [52], Masthoff 2003[53], Montaner et al. 2002 [57], Mukherjee et al. 2001 [58], Noh 2004 [61], Perny and Zucker 1999 [66], Perny and Zucker 2001 [67], Plantie et al. 2005[68], Ricci and Werthner 2002 [73], Sahoo et al. 2006 [79], Schafer 2005 [82], Schickel-Zuber and Faltings 2005 [83], Schmitt et al. 2003 [85], Schmitt et al. 2002 [84], Srikumar and Bhasker 2004 [89], Stolze and Stroebel 2003 [91], Stolze and Rjaibi 2001 [90], Tang and McCalla 2009 [92], Tsai et al. 2006 [96], Yu 2004 [101], Yu 2002 [100], Zimmerman et al. 2004 [103]
优化	Lee and Tang 2007 [43], Price and Messinger 2005 [69], Tewari et al. 2003[93]
级别优先关系	Emi et al. 2003 [22], Nguyen and Haddawy 1999 [60], Nguyen and Haddawy 1999 [59]
其他偏好模型	Ardissono et al. 2003 [5], Cheetham 2003 [14], Lee et al. 2002 [45], Ricci and Nguyen 2007 [74], Wang 2004 [99]

多准则推荐系统的分类依据它们支持的决策问题类型(见表 24.1)、使用的准则类型(见表 24.2)以及全局偏好建模方法(见表 24.3)。基于表 24.1, 有意思的是现有的大多数研究都是关于物品排序的决策问题(如排序候选推荐)。也有一些系统支持依据物品对用户的合适度将物品划分到不同的类别(例如, 推荐物品与非推荐物品)。尽管存在一些应用可以证明相关性, 只有很少的几个系统支持选择和描述问题。此外, 如表 24.2 所示, 被应用的准则族通常是可度量的(measurable), 即用户对物品每个准则进行评分是在可度量范围内的。但是还是有一些系统使用模糊、有序和概率准则来表示候选推荐物品的用户偏好。最后, 表 24.3 显示, 只有很少的多准则推荐涉及全局偏好模型创建, 利用多目标优化和级别优先关系。相反, 大多数使用了价值导向模型, 通常以额外效用函数来预测。也有一些系统没有从多个准则集成预测, 而是使用了原始向量模型作为结果(例如, 通过从所有的准则提供一个评分向量)。

非常重要的一点是现有的系统有时违反了 Roy 方法提出的一致性规则(例如, 没有使用维度的完整集)。此外, 实验结果显示多准则系统即使在没有采用形式化建模方法的时候也有好的满意度。这意味着建模不一致并不总是影响问题性能, 尽管这一问题还有待进一步研究。

24.3 推荐系统的 MCDM 框架：经验教训

根据 MCDM 框架, 在表 24.1~表 24.3 归纳的推荐系统可以认为是多准则推荐系统, 非常重要的一点就是要理解现有推荐系统的类型在这一框架内的位置, 以及 MCDM 框架是否引入新的推荐系统类型。

根据推荐方法通常可以把推荐技术分为以下几类: 基于内容、协同过滤、基于知识和混合推荐[7]。基于内容的推荐技术是根据用户过去喜欢的物品来产生用户的最佳推荐[65], 协同过滤推荐技术是基于其他相似偏好用户的信息来产生推荐[8]。基于知识的方法是利用用户和物品之间的知识来发现满足用户需求的物品[9]。基于知识的推荐的瓶颈

是首先要得到知识库,通过获取知识库来避免冷启动和数据稀疏问题,而这些问题是基于内容和协同过滤仅仅依靠用户的评分所面临的。混合方法是将基于内容、协同过滤和基于知识多种推荐方式进行融合[10]。对 24.2 节总结的典型的 MCDM 推荐系统进行更进一步的分析,我们发现绝大多数推荐系统的多准则特性可以分为以下三类:

- **多属性内容偏好建模**(multi-attribute content preference modeling)。尽管这些系统通常采用单值评分(如数值评分或二进制评分),对于给定的用户,这些系统企图理解和对用户过去喜欢物品的多准则内容的共性建模,然后推荐与用户偏好相匹配的物品。例如,在一个电影推荐系统中,用户偏好电影的这些共性可以表示为流派、演员、导演等用户对电影产生共同喜好的方面。
- **多属性内容搜索与过滤**(multi-attribute content search and filtering)。这类系统允许用户指定基于内容准则的偏好,通过搜索或过滤(比如,只搜索“喜剧”电影或指定“喜剧”电影比“动作”电影更受欢迎),然后给用户推荐与其偏好最相似的物品来满足特定搜索或过滤条件。
- **基于多准则评分的偏好获取**(multi-criteria rating-based preference elicitation)。这些系统允许用户通过多准则评分指定他的个性化偏好(如给电影“Wanted”的故事情节打 2 分,而影视效果打 5 分),然后基于该用户和其他用户多准则评分向用户推荐能够最好地反映用户的个性化偏好的物品。

**多属性内容偏好建模。**对用户偏好建模的一种方法是分析用户购买过或喜欢的物品的多属性内容。很多多准则推荐系统将基于内容的特征直接置入推荐过程中(如使用基于内容的方法),或者是与协同过滤推荐技术融合(如使用混合方法)。在这些系统中,用户通常允许通过单值评分来隐式或显式地表达他们的偏好。利用这些评分,通过分析用户购买或喜欢的物品的个性化内容准则,推荐系统可以自动学习用户基于内容的偏好。例如,为每个用户标识出他喜欢的内容属性(如“喜剧”电影)。因此产生的推荐结果需要考虑这些偏好的内容属性[7]。大量传统的推荐系统采用基于内容、基于知识或者混合推荐方法与多属性偏好建模相结合。人们开发了一些打分或效用函数来基于用户偏好来排序备选物品,其中包括基于检索或基于模型的技术,如贝叶斯分类以及多种机器学习技术[4]。关于这些技术的详细内容请参阅本书第 3 章。

**多属性内容搜索与过滤。**在一些系统中,用户可以明确地指定他们在物品多个属性上的通用性偏好,搜索或过滤技术可以利用这样的偏好信息来找到最相关的物品。例如,在文献[82]中,用户可以标识电影流派、MPAA 评分、他们喜欢的电影长度以及在当前时间哪个属性对于选择电影的决策是最重要的。然后推荐系统通过搜索额外的显性偏好来缩小可选范围。例如,一个用户表示她想看喜剧电影并且电影流派对她来说是最重要的属性,那么只有喜剧会被推荐给她。同样在文献[45]中,用户可以为推荐系统提供不同内容属性的偏好和对应的不同属性的重要性权值。

一些基于知识的推荐系统[35, 37]也可以归入这一类中,因为用户可以通过构建层次分类树来提供他们的通用偏好(例如,对所有的物品特征以层次化方式建模),并为分类树的每个组件分配重要程度水平。因此,推荐系统根据用户预定义的多属性物品分类系统中的偏好,向用户推荐相关度最高的物品。此外,部分基于知识的混合推荐也属于这一类,尤其是基于案例推理的推荐系统,这类推荐系统的物品以结构化的方式表示多准则内容,(例如,使用定义良好的特征集或特征值)[88]。这些系统允许用户在搜索他们感兴趣的物品时指定他们在物品的多维属性上偏好。例如,几个基

于案例的旅行推荐系统[73, 75], 通过多维属性内容上的偏好(如位置、服务、活动)来过滤不想要的物品, 并基于用户的偏好和用户过去的旅行计划或者类似用户的旅行计划对可能的旅行计划进行排序, 从而找到个性化的旅行计划。此外, 还有一些基于案例的推荐系统[9, 70]允许用户评价推荐结果, 通过改进他们的需求作为交互与迭代的推荐过程的一部分, 该方法使用不同的搜索和过滤技术来不断地为用户提供改进的推荐结果。例如, 当用户搜索台式电脑时, 用户可以通过对当前提供的推荐结果(电脑)的个别特征(如价格最便宜)或多个特征(如处理速度快、内存和硬盘容量)进行评论以表达改进后的偏好。

**基于多准则评分的偏好获取。**这类推荐系统引入了多准则评分, 一般通过扩展传统协同过滤方法, 来展示用户对单个物品各个部分的主观偏好。例如, 这类推荐系统不仅允许用户对电影的整体满意度评分, 也可以对电影的不同因素的满意度进行评分, 如影视效果、故事情节或表演。与上面所述系统不同, 不需要用户显性说明他们的偏好或电影视觉效果组件的重要性权值, 而是指出他们喜欢某部电影影视效果的程度。这类系统的典型代表是智能旅行推荐系统[73], 该系统的用户可以对一个旅行打包中的多个物品进行评分, 也可以对整个旅游打包物品评分。然后根据用户评分对候选旅行计划进行排序, 系统通过推荐的旅行计划和用户的需求进行最佳匹配。这类系统或与它相似的多准则评分系统将是本章的重点, 更多原型系统和技术将在后续章节中详述。

总之, 正如以上我们所看到的, 许多使用了传统的基于内容、基于知识和混合技术的推荐系统可以看作多准则推荐系统, 因为它们要么基于用户历史记录, 要么是基于用户指定偏好物品的多属性内容进行偏好建模的。例如, 物品多属性内容的搜索过滤条件(如标识偏好电影的流派或提供多个预定义的流派的偏好值)。然而, 如前面所述, 多准则推荐的最新趋势是研究在协同推荐中引入多准则评分的新颖方法。我们相信这些关于用户偏好的额外信息为提供新颖的推荐结果和创建一个独特的还没有被广泛研究的多准则评分环境提供了很多机会。因此, 在 24.4 节中, 我们综述了这类将单评分和多准则一起使用的系统, 也就是我们所说的多准则评分推荐系统的最新技术。

## 24.4 多准则评分推荐

本节中, 我们通过对与之相对应的单准则评分形式化扩展来定义多准则评分推荐问题, 并进一步探讨额外准则在推荐系统中的优势。

### 24.4.1 传统的单值评分推荐问题

传统的推荐系统运行在用户和物品的二维空间。物品对用户的效用表示为有序集  $R_0$  (如一定范围内的非负整数或实数), 推荐系统的目标是预测某个物品对某个用户的效用。如前面所述, 一个效用函数  $R$  可以形式化地写为

$$R: \text{Users} \times \text{Items} \rightarrow R_0 \quad (24.2)$$

效用函数由用户的输入决定, 比如, 用户对物品的显性数值评分或隐含用户偏好的事务数据(如购买历史)。大多数的传统推荐系统使用单准则评分来表示一个用户对特定物品的整体喜欢程度。(如一个用户对物品的整体效用函数)。例如, 在如表 24.4 所示的一个电影推荐系统中, 用户 Alice 对电影 Wanted 的电影评分是 5(总分是 10), 可以表示为  $R(\text{Alice}, \text{Wanted}) = 5$ 。为了便于呈现, 我们假定采用基于邻居的协同过滤推荐技术进

行评分预测[71]，基于邻居的协同过滤推荐技术是非常流行的基于启发式的推荐技术中的一种。这一技术是基于具有相似偏好的用户(邻居)的评分来预测用户对给定的物品的评分。特别在此例中，推荐系统试图基于观测到的评分来预测 Alice 对电影 Fargo 的评分。由于 Alice 和 John 在以前看过并且评分过的四部电影上有着相似的评分模式(见图 24.4)，为简单起见，这个例子中我们使用 John 对电影 Fargo 的评分(如 9 分)来作为用户 Alice 对电影 Fargo 的预测评分。尽管在实际的推荐系统中，我们会注意到用户通常有不止一个邻居。

表 24.4 单值评分电影推荐系统

		Wanted	WALL-E	Star Wars	Seven	Fargo	
目标用户 →	Alice	5	7	5	7	?	← 要预测的评分
和目标用户最相似 → 的用户	John	5	7	5	7	9	← 用于预测的评分
	Mason	6	6	6	6	5	
	:	:	:	:	:	:	

24.4.2 引入多准则评分来扩展传统推荐系统

随着现实世界中应用数量的增加，在扩展推荐系统时引入多准则评分被认为是下一代推荐系统的一个重要问题[4]。多准则评分系统的例子，包括对餐馆的评分(如食物、装修、服务)的 Zagat's guide，提供用户对电子产品的多准则推荐(如屏幕大小、性能、电池寿命和耗费)的 Buy.com，以及 Yahoo! 电影采用了四个准则(如故事、动作、编导和视觉)以表示用户的评分。由多准则评分(而不是单一整体评分)提供的额外的用户偏好信息可以有效地改善推荐系统的性能。

一些多准则推荐系统在对某个用户对一个指定物品的效用建模时，既采用整体评分  $R_0$ ，也采用用户对每个准则  $c(c=1, \dots, k)$  的评分  $R_1, \dots, R_k$ ；然而有些系统则不使用整体评分，仅仅关注于每个准则的评分。因此，多准则推荐问题的效用公式可以表示为包含整体评分或不包含整体评分，如下所示：

$$R: \text{Users} \times \text{Items} \rightarrow R_0 \times R_1 \times \dots \times R_k \tag{24.3}$$

或

$$R: \text{Users} \times \text{Items} \rightarrow R_1 \times \dots \times R_k \tag{24.4}$$

考虑到每个物品的多准则评分可用性(除了传统的单准则整体评分)，表 24.4 和表 24.5 说明了这些信息对于推荐系统的潜在益处。当 Alice 和 John 在单评分设置中具有相似的偏好时(见表 24.4)，在多准则评分设置中，我们可以看到他们在电影的几个方面观点完全不同(见表 24.5)。进一步检查多准则评分信息，可以发现 Alice 和 Mason 具有相似的评分模式(比 Alice 和 John 更相似)。使用前面的协同过滤方法，考虑多准则评分，基于 Mason 对电影 Fargo 的整体评分为 5，预测 Alice 对电影 Fargo 的整体评分为 5。

这个例子暗示了单准则整体评分可能隐藏了用户对于指定物品的不同侧面偏好的潜在异构性，并且多准则评分可以帮助用户更好地理解用户的偏好，因此可以为用户产生更准确的推荐结果。同时也说明了多准则评分是如何产生更有效的和聚焦的推荐结果，例如，在用户认为故事情节很重要的时候，可以推荐在故事情节上具有更高得分的电影。



表 24.5 多准则电影推荐系统(为每个物品评分：全部、故事、动作、编导和视觉)

目标用户	Wanted	WALL-E	Star Wars	Seven	Fargo	
Alice	5, 2, 2, 8, 8	7, 5, 5, 9, 9	5, 2, 2, 8, 8	7, 5, 5, 9, 9	?, ?, ?, ?, ?	要预测的评分
和目标用户最相似的用户	John	7, 9, 9, 5, 5	5, 8, 8, 2, 2	7, 8, 8, 2, 2	9, 8, 8, 10, 10	
Mason	6, 3, 3, 9, 9	6, 4, 4, 8, 8	6, 3, 3, 9, 9	6, 4, 4, 8, 8	5, 2, 2, 8, 8	用于预测的评分
:	:	:	:	:	:	

因此，推荐系统需要采用多准则评分的新推荐算法和技术。部分系统已经实现了这些算法，我们将在 24.5 节中进行分析。

24.5 多准则评分推荐算法综述

推荐系统通常分两个阶段来计算和产生推荐：

- 预测(prediction)：这一阶段计算用户的偏好。通常这一阶段主要是根据已知的评分或者是其他信息(如用户概况或物品内容)来估计在整个或部分的 User×Item 空间的效用函数  $R$ 。换句话说，就是为未评分物品预测评分。
- 推荐(recommendation)：这个阶段主要是用户通过一些推荐流程来支持用户决策。例如，这个阶段可以向用户推荐最大效用的 top- $N$  物品( $N$  个最高评分的物品)。

多准则评分信息在这两个阶段中以不同的方式应用，已经开发了很多关于预测和推荐的方法。因此，我们可以将现有的多准则评分推荐技术划分为两组，预测阶段使用的技术和推荐阶段使用的技术。下面将分别进行详细阐述。

24.5.1 预测中使用多准则评分

本节对使用多准则评分来预测整体评分或单个准则评分(或两者)的技术进行了综述。通常，推荐技术可以依据效用函数的形成分为两类：基于启发式的，有时也称为基于内存的和基于模型的[4]。基于启发式的技术依据用户观测到的值或依据一定的启发式假设，为用户的每个物品在线计算效用。例如，基于近邻的技术假设两个用户在已观察的物品上有相似的偏好，那么他们在未观测物品上也具有相似偏好。相对来讲，基于模型的技术一般采用统计或机器学习方法来学习一个预测模型。可以更好地解释已观测的数据，然后使用学习到的模型来估计未知物品的效用并产生推荐结果。依据这种分类方式，我们将多准则评分推荐算法也划分为基于启发式的方法和基于模型的方法。

24.5.1.1 启发式方法

已经有一些方法被提出用于扩展传统基于启发式协同技术的相似度计算来反映多准则评分信息[2, 49, 92]。在这些方法中，用户之间的相似度计算是通过聚合每个准则上的相似度或使用多维距离度量的。

特别地，在基于邻居的协同推荐中，是基于具有相似偏好或品味的其他用户的评分来预测指定用户的未知评分。因此，在预测的第一步就是选择一种相似度计算方法以便为每个用户找到他们的邻居集。在单准则评分推荐系统中有多种方法用来计算相似度。最流行的方法是基于相关度和基于余弦的方法。假设  $R(u, i)$  表示用户  $u$  对物品  $i$  的评分， $I(u,$

$u'$ )表示用户  $u$  和用户  $u'$  的共同评分物品。这两种流行的相似度计算方法可以形式化地写为

- 基于 Pearson 相关系数(pearson correlation-based):

$$\text{sim}(u, u') = \frac{\sum_{i \in I(u, u')} (R(u, i) - \overline{R(u)}) (R(u', i) - \overline{R(u')})}{\sqrt{\sum_{i \in I(u, u')} (R(u, i) - \overline{R(u)})^2} \sqrt{\sum_{i \in I(u, u')} (R(u', i) - \overline{R(u')})^2}} \quad (24.5)$$

- 基于余弦(cosine-based):

$$\text{sim}(u, u') = \frac{\sum_{i \in I(u, u')} R(u, i) R(u', i)}{\sqrt{\sum_{i \in I(u, u')} R(u, i)^2} \sqrt{\sum_{i \in I(u, u')} R(u', i)^2}} \quad (24.6)$$

多准则评分推荐系统不能直接使用上面的公式,因为在多准则评分推荐系统中  $R(u, i)$  包含整体评分  $r_0$  和  $k$  维多准则评分  $r_1, \dots, r_k$ , 即  $R(u, i) = (r_0, r_1, \dots, r_k)$ 。因此,每一对  $(u, i)$  都有  $k+1$  个评分值,而不是一个单一的评分。有两种不同的方法可以使用  $k+1$  个评分值来计算相似度。第一种方法是聚合传统的基于单个评分的相似度。这种方法首先采用传统的相似度计算,如相关性或余弦值,分别计算用户在每个准则上的相似度,然后通过聚合  $k+1$  个相似度的值作为用户最后的相似度。Adomavicius 和 Kwon[2]提出两种聚合方法,平均相似度和最坏相似度(如最小值),如式(24.7)和式(24.8)所示。作为一种通用方法, Tang 和 McCalla[92]在论文推荐系统中采用加权和方式计算每篇论文的单个相似度。(如整体评分、附加值、同行推荐程度、学习者的兴趣和背景知识)。如式(24.9)所示,在他们的方法中,  $w_c$  表示每个准则  $c$  的权值,用来反映该准则在推荐结果中的重要性 和有用性。

- 平均相似度(average similarity):

$$\text{sim}_{\text{avg}}(u, u') = \frac{1}{k+1} \sum_{c=0}^k \text{sim}_c(u, u') \quad (24.7)$$

- 最坏(最小值)相似度(worst-case(smallest) similarity):

$$\text{sim}_{\min}(u, u') = \min_{c=0, \dots, k} \text{sim}_c(u, u') \quad (24.8)$$

- 聚合相似度(aggregate similarity):

$$\text{sim}_{\text{aggregate}}(u, u') = \sum_{c=0}^k w_c \text{sim}_c(u, u') \quad (24.9)$$

第二种计算相似度的方法是多维距离度量,如 Manhattan、Euclidean 和 Chebyshev 距离度量[2]。两个用户  $u$  和  $u'$  在物品  $i$  上的距离  $d(R(u, i), R(u', i))$  可以通过下面公式计算:

- Manhattan 距离:

$$\sum_{c=0}^k |R_c(u, i) - R_c(u', i)| \quad (24.10)$$

- Euclidean 距离:

$$\sqrt{\sum_{c=0}^k |R_c(u, i) - R_c(u', i)|^2} \quad (24.11)$$

- Chebyshev 或最大值距离:

$$\max_{c=0, \dots, k} |R_c(u, i) - R_c(u', i)| \quad (24.12)$$

两个用户之间的总体距离可以简单通过计算两个用户共同评分物品的平均距离得到, 可以形式化地写为

$$\text{dist}(u, u') = \frac{1}{|I(u, u')|} \sum_{i \in I(u, u')} d(R(u, i), R(u', i)) \quad (24.13)$$

两个用户越相似(相似度的值越大), 他们之间的距离越小。由于它们之间呈反比关系, 所以需要对两个度量进行一个简单的转换, 如下所示:

$$\text{sim}(u, u') = \frac{1}{1 + \text{dist}(u, u')} \quad (24.14)$$

Manouselis 和 Xostopoulou[49]也提出了三种不同的方法来计算多准则设置中用户之间的相似度: 按优先级相似度(similarity-per-priority)、按评价相似度(similarity-per-evaluation)和按部分效用相似度(similarity-per-partial-utility)。按优先级相似度算法基于用户  $u$  对每个准则  $c$  的重要性权值  $w_c(u)$ (而不是评分  $R(u, i)$ )来计算用户之间相似度。在这种方式中, 通过创建一个与目标用户具有表现相同的偏好性邻居用户集, 然后基于用户的邻居用户的整体效能来预测用户对物品的整体效能。此外, 按评价相似度和按部分效用相似度算法为目标用户在每个准则上分别创建邻居用户, 然后预测每个用户的准则评分。按部分效用相似度算法则是基于用户对每个准则的加权评分来计算相似度。

在这些系统中, 用户之间的相似度是采用多准则评分获取, 剩余的推荐流程和单准则推荐系统的推荐流程是一样的。下一步就是为指定用户找到一个最相似度的邻居集, 然后基于邻居的评分来预测用户对未知物品的评分。因此, 基于相似度的方法仅仅适用于需要计算用户(或物品)相似度的基于邻居的协同过滤推荐技术。

总之, 多准则评分可以通过以下两种方式来计算两个用户之间的相似度[2]: 1) 聚合每个准则的单准则相似度的值; 2) 计算多准则评分在多维空间中的距离。在小规模的 Yahoo! 电影数据集的实验结果显示, 这两种启发式方法都优于传统的单值评分协同过滤技术(即只用单值整体评分), 在 top- $N$  的推荐精度方面提升了 3.8%, 代表了系统为每个用户预测的  $N$  个最相关物品中整体评分真正高的百分比[2]。精度的提升依赖于协同过滤技术中的多个参数, 如邻居规模, top- $N$  推荐的数量。此外, Manouselis 和 Costopoulou [49]建议这些方法可以进一步扩展, 不仅可以通过已知评分信息来计算相似度, 还可以利用每个准则的权值来计算相似度。第二种方法用于电子市场推荐在线应用, 购买者和销售者可以基于用户对多个电子市场的多准则评估, 获得和交换关于价格以及待售商品的信息。与数值平均和随机这类非个性化的算法相比, 使用欧氏距离来度量相似度的按优先级相似度算法的性能最好, 平均绝对误差(MAE)在 1~7 的范围内达到 0.235, 覆盖率(可以推荐给用户的产品范围)达到 93%, 数值平均和随机的 MAE 分别为 0.718 和 2.063, 覆盖率则都达到 100%[49]。

#### 24.5.1.2 基于模型的方法

基于模型的方法是通过已观测到的数据进行学习, 构建了一个预测模型来估计未知评分的。现有的几种多准则推荐系统可以划入此类, 包括聚合函数、概率模型以及多维线性矩阵分解(MSVD)。

**聚合函数方法。**在基于相似度的启发式方法中, 整体评分  $r_o$  通常被简单地作为一个单独的准则评分; 然而在聚合函数方法中, 将整体评分作为多准则评分的一种聚合[2]。根据这一假设, 这一方法需要找到一个聚合函数  $f$  来表示整体评分与多准则评分之间的关系, 即

$$r_0 = f(r_1, \dots, r_k) \quad (24.15)$$

例如, 在一个电影推荐应用中, 故事情节准则的评分有非常高优先权, 也就是说, 故事情节评分高的电影受到某些用户的喜爱, 不管其他准则评分如何。因此, 如果某部电影的故事情节评分预测值高, 为了更精确推荐, 该电影的整体评分也应该高。

聚合函数方法由三步组成, 如图 24.1 所示, 首先, 这一方法可以采用任意的推荐技术来估计  $k$  个单独评分, 也就是  $k$  维多准则评分问题被分解为  $k$  个单准则评分推荐问题。其次, 通过领域专家、统计技术或机器学习技术来选择聚合函数  $f$ 。例如, 领域专家可基于以前的经验和知识建议对每个物品的多准则评分进行简单的平均, 也可以通过统计技术来获取聚合函数, 如线性和非线性回归技术; 还可以使用复杂的机器学习技术, 如人工神经网络。最后, 未评分物品的整体评分是基于  $k$  个单独评分和所选定的聚合函数  $f$  来计算的。

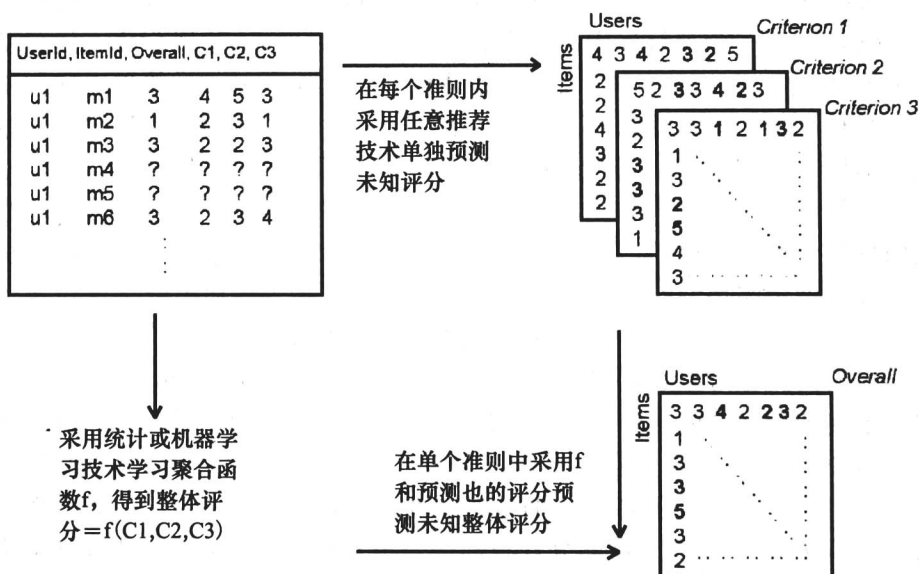


图 24.1 聚合函数方法(三准则评分系统的实例)

尽管前面所述的基于相似度的启发式方法仅应用到基于邻居的协同过滤技术, 通过聚合函数可以将任意一种传统的推荐技术进行融合, 因为在预测的第一步中使用的是单准则评分。作为可能的聚合函数例子, Adomavicius 和 Kwon[2] 基于已知评分使用线性回归和估值系数(如每个准则的权重)。

Adomavicius 和 Kwon[2] 还指出, 聚合函数有不同的范围, 基于整体的(如基于整个数据集学习的单聚合函数)、基于用户的或基于物品的(如根据每一个用户或物品学习一个单独的聚合函数)。

Yahoo! 电影数据的实验分析显示, 利用聚合函数方法(使用多准则评分信息)表现优于传统的单值评分协同过滤技术(只使用单一的整体评分), 在  $\text{top-}N$  ( $N=3, 5, 7$ ) 时候准确度提升  $0.3\% \sim 6.3\%$  [2]。

**概率模型方法。**一些多准则推荐方法采用了数据挖掘和机器学习中越来越流行的概率建模方法。其中一个例子就是 Sahoo 等的工作[79], 它扩展了 Si 和 Jin[86] 的 FMM(易变混合模型, Flexible Mixture Model)。如图 24.2a 所示, FMM 假定有两个隐含变量  $Z_u$  和  $Z_i$  (为用户和物品), 它们被用于确定用户  $u$  对物品  $i$  单值评分。如图 24.2b 所示, Sahoo

等[79]利用 Chow-Liu 树结构发现[16], 还揭示了整体评分  $R_0$  与多准则评分 ( $r_1, r_2, r_3, r_4$ ) 之间的依赖结构, 并将这种依赖结构融入 FMM。

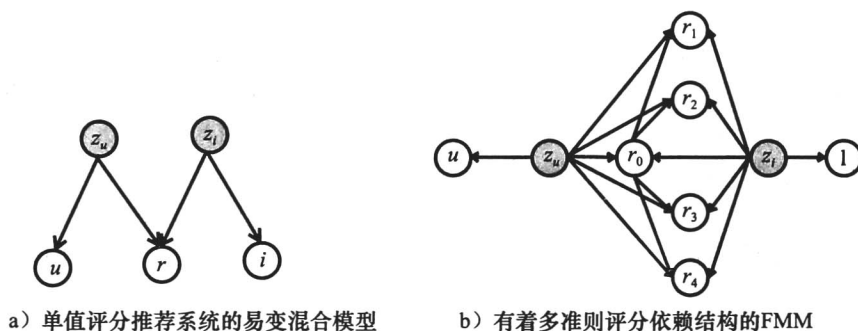


图 24.2 推荐系统中概率模型方法的实例

FMM 方法是基于以下假设, 即三个变量(用户  $u$ 、物品  $i$  和评分  $r$ )的联合概率可以表示为两个隐含变量  $Z_u$  和  $Z_i$  的所有可能的组合的概率和, 如下所示:

$$P(u, i, r) = \sum_{Z_u, Z_i} P(Z_u)P(Z_i)P(u|Z_u)P(i|Z_i)P(r|Z_u, Z_i) \quad (24.16)$$

总之, 目标用户的未知物品的整体评分可以通过以下两步得到: 学习和预测。第一步(学习), FMM 的所有参数都是通过期望最大化算法(EM)来估计得到的; 在第二步(预测)中, 使用估计的参数, 把最有可能的值(如最大概率的评分值)作为未知物品的整体预测评分, 这种方法已经扩展到多准则评分, 具体的算法可以在文献[79]找到。

Sahoo 等[79]将他们在图 24.2b 中的模型与假设的隐含变量条件独立的评分模型进行比较, 并发现有依赖结构的模型比独立假设的模型效果要好。这一发现揭示了在多准则评分中存在晕轮效应(halo effect)。晕轮效应是心理学中经常研究的一种现象, 即对属于某一类的物体的认知偏见会影响对其他类的认知[94]。在多准则推荐系统中, 用户的单准则评分之间因为晕轮效应会产生相互关联, 而每个单准则评分和整体评分的关联程度比对单个准则之间的影响程度更大[79]。换句话说, 用户对特定物品的整体评分会影响其对单个准则的评分。因此, 控制整体评分并减小它的晕轮效应, 并且每个评分能够互相独立, 如图 24.2b 的 Chow-Liu 树形依赖结构所示。

使用 Yahoo! 电影数据集, Sahoo 等[79]认为, 多准则评分信息在可用的训练数据非常少时, (如用于训练的数据不到整个数据集的 15%)是具有优势的。另一个方面, 当有大量的训练数据可用时, 额外的评分信息没带来更多提升。在这一分析中, 他们采用 MAE 来度量推荐结果的准确度。然而, 当他们在检索 TopN 物品中使用准确率和召回率来验证概率模型方法, 他们的模型在所有例子中都有较好的表现(包括大数据集和小数据集), 最大提升了 10%。随着更多的训练数据, 多准则评分和单准则评分在准确率和召回率方面的差异在逐渐缩小。

**多维线性矩阵分解方法(MSVD)。**Li 等[46]提出了一种利用 MSVD 技术来改善传统协同过滤的算法。奇异值分解(Singular Value Decomposition, SVD)技术在线性代数中得到广泛研究, 由于它可以有效地提升推荐结果的准确度, 使得它在推荐系统中的使用也越来越广泛[26, 41, 81]。在单值评分推荐系统中, 这类技术发现了物品的隐含特征, 包括明确定义的物品维度和无法解释的维度。特别地, 是使用  $k$  个隐含特征(如 Rank-K SVD), 用户  $u$  与用户因子向量  $p_u$  (用户对特征  $k$  的偏好), 物品  $i$  与物品因子向量  $q_i$  (物品在特征  $k$



上的权重)相关联。在用户和物品因子向量中的所有值都被估计后,用户  $u$  对物品  $i$  的偏好程度可以表示为  $R^*(u, i)$ , 可以通过两个向量的内积得到, 即

$$R^*(u, i) = P_u^T q_i \quad (24.17)$$

SVD 技术被经常用于二维数据的分解, 而 MSVD 技术[17]可以用于多维数据的分解, 如多准则评分。尤其是 Li 等在[46]中使用 MSVD 降低多准则评分数据的维度, 并在餐馆推荐系统的情境下评估了它们的方法, 在餐馆推荐系统中用户可以在十个准则(如菜品、环境、服务等)对一个餐馆进行评分。更确切地说, 他们使用 MSVD 技术来发掘用户、物品和准则三者之间的关系, 并使用这些数据来发掘每个用户的邻居以及计算 topN 推荐。结果显示, 与传统的单值模型相比, 他们的方法在推荐结果的准确度上(topN 准确度方式进行度量)提升了 5%。

总之, 上面的方法采用一些复杂的学习技术来处理多准则推荐做了一些尝试。我们希望在将来能看到更多类似的技术。

在 24.5.2 节中, 我们探讨向用户推荐物品的不同方法, 我们假设未知的多准则评分采用上述任意一种方法已经被估算出来。

## 24.5.2 推荐中使用多准则评分

如前所述, 多准则推荐系统可能选择建模一个用户对一个物品的效能, 可以同时包括整体评分和单个物品准则评分, 也可以只包括单个准则评分。如果整体评分作为模型的一部分, 推荐流程就非常直接, 推荐系统使用整体评分来向用户推荐最高预测值的物品。换句话说, 推荐流程与传统的单准则推荐系统基本上相同。

然而, 没有整体评分时, 推荐流程就会更复杂, 因为如何建立物品的整体序列不那么明显。例如, 假定有个双准则电影推荐系统, 用户依据电影的故事情节和影视效果来判断。在以下两个物品中选择一个电影来推荐: 1) 电影 X, 预测的故事情节为 8, 影视效果为 2; 2) 电影 Y, 预测的故事情节为 5, 影视效果为 5。由于没有整体评分来对电影排序, 除非采用其他的建模方法, 用非数值方式(如基于规则的)来表示用户偏好, 否则很难判断哪部电影更好。在推荐系统文献中, 提出了几种方法来应对这个问题: 有些试图为每个用户设计一个物品序列和一个单值全局优化方案, 另一部分则采用部分物品序列找到多个解(帕雷托最优)。下面我们简单介绍与多准则优化的相关工作, 描述一些已经用于推荐系统的方法, 以及探讨在多准则评分的推荐阶段中的其他潜在应用方法。

### 24.5.2.1 相关工作: 多准则优化

多准则优化问题在操作研究中广泛应用[21], 而不是在推荐系统的背景下。多准则优化问题帮助决策者在多个准则冲突或互相竞争的情况下选择最佳选项。例如, 从多个视角, 如金融、人力资源相关以及环境方面都可以纳入组织决策中。如文献[4]讨论, 以下方法主要用于处理多准则优化问题, 也可以应用到推荐系统:

- 找到帕雷托最优解;
- 通过多准则线性融合, 将多准则优化问题简化为单准则推荐优化问题;
- 优化最重要的准则, 把其他准则作为约束;
- 连续地每次优化一个准则, 转换为最优解来约束, 或在其他准则上重复这一流程。

下面将描述在推荐系统中应用的几种推荐方法, 所有这些方法都源于多准则优化技术。

### 24.5.2.2 设计物品推荐整体序列

在推荐系统的文献中, 部分采用决策科学中多属性效用理论, 这个可以被作为一种多

准则的线性组合并找到最优解[42]。

例如, Lakiotaki 等[42]提出的方法采用了文献[87]提出 UTA 方法对物品进行排序。他们的算法的目标是通过加入每个准则  $c(c=1, \dots, k)$  边界效用来为每个用户估计特定物品的整体效用。

$$U = \sum_{c=1}^k u_c(g_c) \quad (24.18)$$

其中,  $g_c$  是对准则  $c$  的评分,  $u_c(g_c)$  是特定用户的非增的实数函数(边界效用函数)。由于这一模型使用有序回归模型技术进行排序, Kendall' tau 被用于度量两个定序水平变量的相关性来比较实际顺序和预测顺序。使用 Yahoo! 数据集的实验结果显示, 20.4% 的用户得到 Kendall's tau 为 1, 即推荐系统的预测序列与用户声明序列的一致。所有用户的 Kendall's tau 是 0.74。他们的模型也使用了接受者操作曲线(Receiver Operating Curve, ROC)进行评估, ROC 是用于描述真正(true positive)与假正(false positive)之间的相对权衡。得到的曲线下区域(Area Under Curve, AUC)是 0.81, 其中 1 表示完美分类器, 0.5 表示随机分类器的性能, 结果显示多准则推荐系统在用户偏好建模方面有显著的改进。

相类似, Manourselis 和 Costopoulou[49]中提出一种方法, 整体效用  $U$  要么是将  $k$  个预测部分效用求和, 要么是根据用户对每个准则  $c$  的权值  $w_c$  对预测评分进行加权求和。这两种情况中, 候选物品的整体效用可用以下聚合函数来计算:

$$U = \sum_{c=1}^k u_c = \sum_{c=1}^k w_c r_c \quad (24.19)$$

最后, 利用如上所述的任意技术建立起候选物品的整体序列, 每个用户都得到最大效用的推荐物品。

### 24.5.2.3 发现帕雷托最优物品推荐

当多个物品相关联的多个准则冲突和物品整体序列不是可以直接获取时, 这一方法可以从大量的备选物品中找到几个好的物品。数据包络分析(Data Envelopment Analysis, DEA), 通常也被称为边界分析, 在操作研究中广泛用于度量决策单元的产出效果(Decision Making Units, DMU)[13]。DEA 计算用来发现哪些物品是“最佳”的效率边界。DEA 不需要为每个准则定制先验权值, 它利用线性规划为每个 DMU 得到最佳权值集合。尤其在多准则推荐系统中, 假定所有的备选物品都可以推荐给用户(包括所有准则的预测评分), DEA 能够决定包括在所有准则上有最好评分的候选物品并作为缩减后的物品集, 这些物品可以推荐给用户。

尽管 DEA 没有被直接应用到多准则评分推荐中, 但是非常相似的是, 缺少整体评价的多准则推荐问题也可以表述成一个在数据库领域的数据库查询问题[43]。Lee 和 Teng[43]使用 Skyline 查询, 利用多准则(如食物、装饰、服务和消费), 来寻找最好的餐馆。如图 24.3 中所示, Skyline 查询标识了一些 skyline 点(即帕雷托最优点), 这些点并不依赖于其他的备选餐馆的二维数据空间(食物和装饰)。对于一个给定用户, 如果存在另外一个备选物品在所有准则上的评分都优于它, 这个候选物品就被认为是最适合推

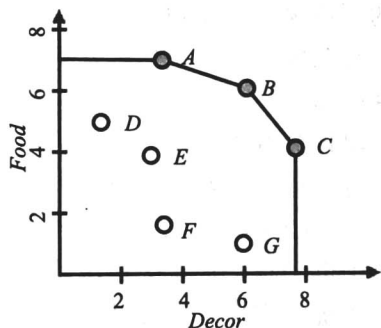


图 24.3 二维空间中 skyline 点(最佳候选餐馆)的实例

荐的。

Zagat Suvery 在[43]使用多准则评分的实验结果显示,使用 Skyline 查询的推荐系统可以减少选择的数量。例如,当一个用户搜索位于纽约的自助餐餐馆,且价格低于 30 美元,基于四个准则,系统在 12 个备选餐厅中系统推荐了 2 个。然而,这一基础工作需要多个方向进一步扩展,随着准则个数的增加 Skyline 查询不能很好地扩展,因为在大量的 Skyline 点情况时导致很高的计算代价。

#### 24.5.2.4 使用多准则评分作为推荐过滤器

与在推荐系统中使用内容属性来对推荐结果过滤[45, 82]类似,多准则评分也可以被用于对推荐结果进行过滤。例如,一个用户在某一特定时间指定只有故事情节很好的电影才能推荐给她,而不管其他准则,那么在推荐时只有故事情节上预测评分( $\geq 9$ , 满分 10 分)的电影才推荐给该用户。这一方法和内容推荐[45, 82],以及情境感知[3]中推荐方法对推荐内容的过滤方式相似;然而,又与它们有一些细微不同,因为它的过滤方式不是依据客观的内容属性(如电影时间 $< 120$ 分钟)或额外情境信息维度(如 TimeofWeek = Weekend),而是基于主观的评分准则(如故事情节 $\geq 9$ )来过滤的,其预测结果主要依赖于用户的口味和偏好。

### 24.6 讨论及未来工作

推荐系统是一个充满生机与不断变化的研究领域。在最近一些重要的研究中,推荐系统开始采用用户提供的多准则评分。本章研究了多准则推荐系统的算法和技术。这些新系统还没有被广泛研究,本节对这类推荐系统的一系列挑战和研究方向进行阐述。

**应对干预。**多准则评分提供的额外信息可能导致一个非常重要的问题就是干预。为了使推荐系统取得好的推荐性能,通常需要用户提供一定数量的偏好反馈(如物品评分)。这一问题在单评分推荐系统中同样存在,一些在用户偏好反馈获取中降低干预性的方法被提出[40, 56, 62]。由于用户需要对物品的多个属性进行评分,所以多准则评分系统需要更高层次的用户参与。因此,采用度量多属性评分的代价和收益变得异常重要,需要在用户需求 and 系统设计需求之间找到一个最佳解决方案。基于以前一系列的决策可以采用偏好分解方法来形成隐含的偏好模型。一个典型的例子就是 UTA(即效用递增, Utility Additive)方法,可以根据已知评分来抽取效用函数[42]。另外一个例子就是根据用户在多属性上的评论来获取用户的偏好,从而最小化干预性[1, 68]。也有一些经验方法具有较低的算法复杂度[80]。最后,在多准则推荐系统中可以进一步研究用户提交多个评分对用户满意度的影响。

**单评分推荐技术的复用。**在过去 10~15 年里,人们提出了大量关于单评分推荐的技术。有一些技术经过拓展后可以应用于多准则推荐系统。例如,Manouselis 和 costopoulou[51]给出的大量设计建议选项,基于邻居的系统过滤技术可以被多准则评分采用。另外一个例子是,近年来提出了大量的复杂混合推荐方法[11],其中有一些也可以在多准则推荐系统中应用。另外其他一些更复杂的技术,如基于 DEA 或多准则优化,都可以被采用并且被拓展到多准则评分环境中选择最优的物品。

**预测相对偏好。**另外一种定义多准则推荐问题的方法可以是将问题定义成预测用户的相对偏好,而不是预测绝对评分值。有一些研究采用基于顺序(Order-based)技术来构建物品的相对顺序。例如, Freund 等[25]在知名的 AdaBoost 方法基础上开发了在多准则环境

下的 RankBoost 算法,这类算法可以通过对某一用户不同评分准则下获取的相对顺序进行聚合。特别地,在 DIVA 系统中也采用了这一方法[59, 60]。

**构建物品评价标准。**我们需要更多地研究如何选择和构建一个最佳的准则集来评价一个物品。例如,目前大多数的多准则推荐系统需要用户对一个物品的多个属性进行评分。这些单一层次的评价标准可以进一步细化为子准则,划分的层次取决于给定的问题。例如,在一个电影推荐系统中,电影的特效可以分为声音特效和图像特效。多级准则的信息可以帮助更好地理解用户偏好,并且多种方法,如层次化分析过程(AHP)可以被用于考虑准则的层次[78],Schmitt 等[85]在它们的系统中采用这个方法。我们在考虑物品的多准则时需要仔细审视准则之间的相关性,因为准则的选择对推荐质量的影响很大。此外,如前所述,拥有一致的准则簇对一个特定的推荐系统是非常重要的,因为这样准则才能保持单调、完整和不冗余。总之,构建一个推荐问题的准则集是一个非常有趣和重要的研究主题。

**应对多准则评分缺失。**与单评分推荐系统相比,多准则推荐系统需要用户提供更多的数据,无形中增加了数据缺失或不完整的可能性。应对数据缺失最常用的方法是期望值最大(expectation maximization)算法[18],为不完整数据找到最大可能性的估值。文献[79]提出了一种概率建模方法来使用 EM 算法对多准则评分中缺失的评分机型预测。其他技术的可用性需要进一步探索,同时也需要研究新的技术来应对多准则的特性,如准则间的可能存在依赖关系。

**多准则环境下的群组推荐。**一些面向群组推荐的技术也可用于多准则环境中。根据文献[31],通过聚合部分用户的多样性偏好得到群组偏好模型,同样,在多准则评分中的用户对物品的偏好也可以通过聚合不同准则的偏好获得。更确切地说,在个性化偏好聚合过程中可能存在不同的目标[55, 63],例如,用户平均满意度最大化、不满意度的最小化或提供一定程度的公平性。多准则评分推荐系统也可以研究采用一些多准则偏好的聚合方法。

**开发新的 MCDM 建模方法。**从 MCDM 的角度来讲,推荐问题使决策建模者面临新的挑战。一方面,有很多技术可以应用到该系统中,例如,文献[68]中提出的算法中的敏感度分析;另一方面,一些研究表明,推荐不是一个单一决策问题,因为它需要对几个决策问题同时处理,每个人都会影响对其他人的推荐[54]。既不是典型的群组决策,也不是个体协商[66]。因此,需要提出新的 MCDM 建模方法,并在多准则推荐系统中测试[20]。

**大规模的多准则评分数据的收集。**可以用来进行算法测试和参数化的多准则数据集非常少。如果想在推荐系统的这一新领域取得成功,为学术界提供一些标准的现实世界的多准则评分数据集是非常重要的。一种面向标准化表示,复用和互操作的多准则评分数据集在电子学习领域中已经有了初步的尝试[98]。

本节讨论了多准则推荐系统领域可能的几个未来研究方向,而推荐系统学术界对多准则推荐系统应该是非常感兴趣的,这个列表并不是非常详尽,我们认为该领域的研究仅仅处于起步阶段,还有大量的课题可以被研究和探索,以求推动多准则推荐系统研究。

## 24.7 总结

本章的主要目的是对多准则推荐系统进行综述。首先,我们将推荐问题定义为多准则决策问题(MCDM),并综述了支撑多准则推荐系统实现的 MCDM 方法和技术。接着,我们聚焦于多准则评分推荐技术,这一技术通过建模用户对每个物品的效用作为一个评分向

量与其他属性一起产生推荐,并总结了当前采用的一些多准则评分预测和产生推荐的技术,文章最后讨论了多准则推荐系统的一些尚未解决的问题及未来的挑战。

## 致谢

本章对多准则推荐系统进行了系统性的回顾,以及相关工作的路线图描述,讨论了一些非常有前景的研究方向。然而我们认为这仅仅是在探索推荐系统这一广泛领域的第一步,想要开启多准则推荐的全部潜力,还有很多问题需要进一步研究和解决。

## 参考文献

1. Aciar, S., Zhang, D., Simoff, S., Debenham, J. Informed Recommender: Basing Recommendations on Consumer Product Reviews. *IEEE Intelligent Systems*, 22(3):39–47, 2007.
2. Adomavicius, G., Kwon, Y. New Recommendation Techniques for Multi-Criteria Rating Systems. *IEEE Intelligent Systems*, 22(3):48–55, 2007.
3. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.
4. Adomavicius, G., Tuzhilin, A. Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
5. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P. Intrigue: Personalised Recommendation of Tourist Attractions for Desktop and Handset Devices. *Applied Artificial Intelligence*, Special Issue on Artificial Intelligence for Cultural Heritage and Digital Libraries, 17(8):687–714, 2003.
6. Ariely, D., Lynch, J.G.Jr., Aparicio, M. Learning by Collaborative and Individual-based recommendation Agents. *Journal of Consumer Psychology*, 14(1&2):81–95, 2004.
7. Balabanovic, M., Shoham, Y. Fab:Content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, 1997.
8. Breese, S., Heckerman, D., Kadie, C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proc. of the 14th Conference on Uncertainty in Artificial Intelligence*, volume 461, pages 43–52. San Francisco, CA, 1998.
9. Burke, R. Knowledge-Based Recommender Systems. *Encyclopedia of Library and Information Systems*, 69(Supplement 32):175–186, 2000.
10. Burke, R. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.
11. Burke, R. Hybrid Web Recommender Systems. In P. Brusilovsky, A. Kobsa, W., Nejdl (Eds.), *The Adaptive Web: Methods and Strategies of Web, Personalization, Lecture Notes in Computer Science*, 4321:377–408, 2007.
12. Cantador, I., Fernandez, M., Castells, P. A Collaborative Recommendation Framework for Ontology Evaluation and Reuse. In *Proc. of the International ECAI Workshop on Recommender Systems*. Riva del Garda, Italy, 2006.
13. Charnes, A., Cooper, W., Rhodes, E. Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2(6):429–444, 1978.
14. Cheetham, W. Global Grade Selector: A Recommender System for Supporting the Sale of Plastic Resin. *Technical Information Series*, GE Global Research, TR 2003GRC261, 2003.
15. Choi, S.H., Cho, Y.H. An utility range-based similar product recommendation algorithm for collaborative companies. *Expert Systems with Applications*, 27(4):549–557, 2004.
16. Chow, C., Liu, C. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.
17. De Lathauwer, L., De Moor, B., Vandewalle, J. A Multilinear Singular Value Decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.
18. Dempster, A., Laird, N., Rubin, D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
19. Denguir-Rekik, A., Montmain, J., Mauris, G. A fuzzy-valued Choquet-integral-based multi-criteria decision-making support for marketing and benchmarking activities in e-commerce organizations. In *Proc. of the MCDM 2006*. Chania, Greece, 2006.



20. DIMACS/LAMSADE. Computer science and decision theory: Applications of notions of consensus description of the research issues in the project. In *COST Action ICO602*. <http://dimacs.rutgers.edu/Workshops/Lamsade/propmod.pdf>, 2004.
21. Ehrgott, M. *Multicriteria Optimization*. Springer Verlag, 2005.
22. Emi, Y., Suetoshi, E., Shinohara, I., Toshikazu, K. Development of a recommendation system with multiple subjective evaluation process models. In *Proc. of the 2003 International Conference on Cyberworlds (CW03) 0-7695-1922-9/03*. Singapore, 2003.
23. Falle, W., Stoeffler, D., Russ, C., Zanker, M., Felfernig, A. Using knowledge-based advisor technology for improved customer satisfaction in the shoe industry. In *Proc. of International Conference on Economic, Technical and Organisational aspects of Product Configuration Systems*, Technical University of Denmark, Copenhagen, 2004.
24. Figueira, J., Greco, S., Ehrgott, M. *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer Verlag, 2005.
25. Freund, Y., Iyer, R., Schapire, R.E., Singer, Y. An efficient boosting algorithm for combining preferences. *The Journal of Machine Learning Research*, 4:933–969, 2003.
26. Funk, S. Netflix update: Try this at home. <http://sifter.org/simon/journal/20061211.html>, 2006.
27. Ghosh, S., Munde, M., Hernandez, K., Sen, S. Voting for movies: the anatomy of a recommender system. In *Proc. of the 3rd Annual Conference on Autonomous Agents*, pages 434–435. ACM, 1999.
28. Guan, S., Ngoo, C.S., Zhu, F. Handy broker: an intelligent product-brokering agent for m-commerce applications with user preference tracking. *Electronic Commerce Research and Applications*, 1(3-4):314–330, 2002.
29. Herrera-Viedma, E., Pasi, G., Lopez-Herrera, A.G. Evaluating the Information Quality of Web Sites: A Qualitative Methodology Based on Fuzzy Computing with Words. *Journal of the American Society for Information Science and Technology*, 57(4):538–549, 2006.
30. Jacquet-Lagrèze, E., Siskos, Y. Preference disaggregation: 20 years of mcda experience. *European Journal of Operational Research*, 130(2):233–245, 2001.
31. Jameson, A., Smyth, B. Recommendation to groups. Brusilovsky P., Kobsa A., and Nejdl W., eds. *Adaptive Web: Methods and Strategies of Web Personalization*, Springer, Berlin, 596–627, 2007.
32. Karacapilidis, N., Hatzieleftheriou, L. A hybrid framework for similarity-based recommendations. *International Journal of Business Intelligence and Data Mining*, 1(1):107–121, 2005.
33. Keeney, R.L. *Value-focused Thinking: A Path to Creative Decisionmaking*. Cambridge MA: Harvard Univ Press, 1992.
34. Keeney, R.L., Raiffa, H. *Decisions with Multiple Objectives: Preferences and Value Trade-offs*. Cambridge Univ Press, 1976.
35. Kerschberg, L., Kim, W., Scime, A. Websifter /uppercaseII: A Personalizable Meta-Search Agent based on Semantic Weighted Taxonomy Tree. In *Proc. of the International Conference on Internet Computing*, pages 14–20. Las Vegas, Nevada, 2001.
36. Kim, T.H., Yang, S.B. Using Attributes to Improve Prediction Quality in Collaborative Filtering. In *Proc. of the E-Commerce and Web Technologies: 5th International Conference, EC-Web 2004*, Zaragoza, Spain, 2004.
37. Kim, W., Kerschberg, L., Scime, A. Learning for automatic personalization in a semantic taxonomy-based meta-search agent. *Electronic Commerce Research and Applications*, 1(2):150–173, 2002.
38. Kleinberg, J., Sandler, M. Convergent Algorithms for Collaborative Filtering. In *Proc. of the 4th ACM Conference on Electronic Commerce*, pages 1–10. San Diego, CA, 2003.
39. Konstan, J.A. Introduction to Recommender Systems: Algorithms and Evaluation. *ACM Transactions on Information Systems (TOIS)*, 22(1):1–4, 2004.
40. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J. Grouplens: Applying Collaborative Filtering to Usenet News. *Communications of the ACM*, 40(3):77–87, 1997.
41. Koren, Y. Collaborative Filtering with Temporal Dynamics. In *Proc. of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 447–456. ACM New York, NY, USA, 2009.
42. Lakiotaki, K., Tsafarakis, S., Matsatsinis, N. UTA-Rec: A Recommender System Based on Multiple Criteria Analysis. In *Proc. of the 2008 ACM Conference on Recommender Systems*, pages 219–226. ACM New York, NY, USA, 2008.
43. Lee, H.H., Teng, W.G. Incorporating Multi-Criteria Ratings in Recommendation Systems. In

- IEEE International Conference on Information Reuse and Integration*, pages 273–278, 2007.
44. Lee, W.P. Towards agent-based decision making in the electronic marketplace: interactive recommendation and automated negotiation. *Expert Systems with Applications*, 27(4):665–679, 2004.
  45. Lee, W.P., Liu, C.H., Lu, C.C. Intelligent agent-based systems for personalized recommendations in Internet commerce. *Expert Systems with Applications*, 22(4):275–284, 2002.
  46. Li, Q., Wang, C., Geng, G. Improving Personalized Services in Mobile Commerce by a Novel Multicriteria Rating Approach. In *Proc. of the 17th International World Wide Web Conference*. Beijing, China, 2008.
  47. Liu, D.R., Shih, Y.Y. Integrating AHP and data mining for product recommendation based on customer lifetime value. *Information & Management*, 42(3):387–400, 2005.
  48. Manouselis, N., Costopoulou, C. Analysis and Classification of Multi-Criteria Recommender Systems. *World Wide Web: Internet and Web Information Systems*, 10(4):415–441, 2007.
  49. Manouselis, N., Costopoulou, C. Experimental Analysis of Design Choices in Multi-Attribute Utility Collaborative Filtering. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(2):311–332, 2007.
  50. Manouselis, N., Costopoulou, C. Towards a Design Process for Intelligent Product Recommendation Services in e-Markets. *Artificial Intelligence and Integrated Information Systems: Emerging Technologies and Applications*, pages 398–417, 2007.
  51. Manouselis, N., Costopoulou, C. Overview of design options for neighborhood-based collaborative filtering systems. *Personalized Information Retrieval and Access: Concepts, Methods and Practices*, pages 30–54, 2008.
  52. Manouselis, N., Sampson, D. A Multi-criteria Model to Support Automatic Recommendation of e-Learning Quality Approaches. In *Proc. of the 16th World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA)*. Lugano, Switzerland, 2004.
  53. Masthoff, J. Modeling the Multiple People That Are Me. In *Proc. of the International Conference on User Modelling (UM2003)*, pages 258–262. Johnstown, USA, 2003.
  54. Matsatsinis, N.F., Samaras, A.P. MCDA and preference disaggregation in group decision support. *European Journal of Operational Research*, 130(2):414–429, 2001.
  55. McCarthy, J. Pocket RestaurantFinder: A situated recommender system for groups. In *Proc. of the Workshop on Mobile Ad-Hoc Communication at the 2002 ACM Conference on Human Factors in Computer Systems*. Minneapolis, MN, 2002.
  56. Middleton, S.E., Shadbolt, N.R., Roure, D.C. Ontological user profiling in recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):54–88, 2004.
  57. Montaner, M., Lopez, B., de la Rosa, J.L., Opinion-Based Filtering Through Trust. In *Proc. of the 6th International Workshop on Cooperative Information Agents VI*, LNCS 2446, pages 164–178, Springer-Verlag, 2002.
  58. Mukherjee, R., Dutta, P.S., Jonsdottir, G., Sen, S. MOVIES2GO: An Online Voting Based Movie Recommender System. In *Proc. of the 5th International Conference on Autonomous Agents*, pages 114–115. Montreal, Canada, 2001.
  59. Nguyen, H., Haddawy, P. DIVA: Applying Decision Theory to Collaborative Filtering. In *Proc. of the AAAI Workshop on Recommender Systems*. Madison, WI, 1998.
  60. Nguyen, H., Haddawy, P. The Decision-Theoretic Video Advisor In *Proc. of the 15th Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pages 494–501. Stockholm, Sweden, 1999.
  61. Noh, S. Implementing Purchasing Assistant Using Personal Profile. In *Proc. IADIS International Conference on Applied Computing*. Lisbon, Portugal, 2004.
  62. Oard, D., Kim, J. Modeling information content using observable behavior. In *Proc. of the Annual Meeting-American Society for Information Science*, volume 38, pages 481–488. Washington DC., 2001.
  63. O'Connor, M., Cosley, D., Konstan, J.A., Riedl, J. PolyLens: A Recommender System for Groups of Users. In *Proc. of the 7th Conference on European Conference on Computer Supported Cooperative Work*, pages 199–218. Kluwer Academic Publishers, 2001.
  64. Pardalos, P., Siskos, Y., Zopounidis, C. *Advances in Multicriteria Analysis*. Dordrecht: Kluwer Academia Publishers, 1995.
  65. Pazzani, M., Billsus, D. Learning and Revising User Profiles: The Identification of Interesting Web Sites. *Machine Learning*, 27(3):313–331, 1997.
  66. Perny, P., Zucker, J.D. Collaborative Filtering Methods based on Fuzzy Preference Relations. In *Proc. of EUROFUSE-SIC 99*, pages 279–285, 1999.
  67. Perny, P., Zucker, J.D. Preference-based Search and Machine Learning for Collaborative

- Filtering: the Film-Conseil Movie Recommender System. *Information, Interaction, Intelligence*, 1(1):1–40, 2001.
68. Plantie, M., Montmain, J., Dray, G. Movies Recommenders Systems: Automation of the Information and Evaluation Phases in a Multi-criteria Decision-Making Process. In *Proc. of the DEXA 2005*, LNCS 3588, pages 633–644. Berlin Heidelberg: Springer-Verlag, 2005.
  69. Price, B., Messinger, P.R. Optimal Recommendation Sets: Covering Uncertainty over User Preferences. *Proc. of the National Conference on Artificial Intelligence*, 2005.
  70. Reilly, J., McCarthy, K., McGinty, L., Smyth, B. Incremental Critiquing. *Knowledge-Based Systems*, 18(4-5):143–151, 2005.
  71. Resnick, P., Iacovou, N., Sushak, M., Bergstrom, P., Riedl, J. GroupLens: An open architecture for collaborative filtering of netnews. In *Proc. of the 1994 ACM Conference on Computer Supported Cooperative Work*, pages 175–186, 1994.
  72. Resnick, P., Varian, H.R. Recommender Systems. *Communications of the ACM*, 40(3):56–58, 1997.
  73. Ricci, F., Arslan, B., Mirzadeh, N., Venturini, A. ITR: A Case-Based Travel Advisory System. *Advances in Case-Based Reasoning*, LNAI 2416, pages 613–627. Springer-Verlag: Berlin Heidelberg, 2002.
  74. Ricci, F., Nguyen, Q.N. Acquiring and Revising Preferences in a Critique-Based Mobile Recommender System. *IEEE Intelligent Systems*, 22(3):22–29, 2007.
  75. Ricci, F., Venturini, A., Cavada, D., Mirzadeh, N., Blaas, D., Nones, M. Product Recommendation with Interactive Query Management and Twofold Similarity. *Proc. of the 5th International Conference on Case-Based Reasoning*, Trondheim, Norway, 2003.
  76. Roy, B., Bouyssou, D. *Aide multicritère à la décision: méthodes et cas*. Economica Paris, 1993.
  77. Roy, B., McCord, M.R. *Multicriteria Methodology for Decision Aiding*. Springer, 1996.
  78. Saaty, T.L. *Optimization in Integers and Related Extremal Problems*. McGraw-Hill, 1970.
  79. Sahoo, N., Krishnan, R., Duncan, G., Callan, J.P. Collaborative Filtering with Multi-component Rating for Recommender Systems. In *Proc. of the 16th Workshop on Information Technologies and Systems*. Milwaukee, WI, 2006.
  80. Sampaio, I., Ramalho, G., Corruble, V., Prudencio, R. Acquiring the Preferences of New Users in Recommender Systems: The Role of Item Controversy. In *Proc. of the 17th European Conference on Artificial Intelligence (ECAI) Workshop on Recommender Systems*, pages 107–110. Riva del Garda, Italy, 2006.
  81. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J. Application of Dimensionality Reduction in Recommender System - A Case Study. In *Proc. of the Workshop on Knowledge Discovery in the Web (WebKDD)*, 2000.
  82. Schafer, J.B. DynamicLens: A Dynamic User-Interface for a Meta-Recommendation Systems. In *Proc. of the Workshop on the Next Stage of Recommender Systems Research at the ACM Intelligent User Interfaces Conference*, 2005.
  83. Schickel-Zuber, V., Faltings, B. Heterogeneous Attribute Utility Model: A new approach for modeling user profiles for recommendation systems. In *Proc. of the Workshop on Knowledge Discovery in the Web (WebKDD)*. Chicago, Illinois, 2005.
  84. Schmitt, C., Dengler, D., Bauer, M. The MAUT-Machine: An Adaptive Recommender System. In *Proc. of the Workshop on Adaptivitat und Benutzermodellierung in Interaktiven Softwaresystemen (ABIS)*. Hannover, Germany, 2002.
  85. Schmitt, C., Dengler, D., Bauer, M. Multivariate Preference Models and Decision Making with the MAUT Machine. In *Proc. of the 9th International Conference on User Modeling (UM 2003)*, pages 297–302, 2003.
  86. Si, L., Jin, R. Flexible Mixture Model for Collaborative Filtering. In *Proc. of the 20th International Conference on Machine Learning*, volume 20, pages 704–711. AAAI Press, 2003.
  87. Siskos, Y., Grigoroudis, E., Matsatsinis, N.F. *UTA Methods*. Springer, 2005.
  88. Smyth, B. Case-Based Recommendation. *The Adaptive Web: Methods and Strategies of Web Personalization*, 2007.
  89. Srikumar, K., Bhasker, B. Personalized Product Selection in Internet Business. *Journal of Electronic Commerce Research*, 5(4):216–227, 2004.
  90. Stolze, M., Rjaibi, W. Towards Scalable Scoring for Preference-based Item Recommendation. *IEEE Data Engineering Bulletin*, 24(3):42–49, 2001.
  91. Stolze, M., Stroebel, M. Dealing with Learning in eCommerce Product Navigation and Decision Support: The Teaching Salesman Problem. In *Proc. of the 2nd Interdisciplinary World Congress on Mass Customization and Personalization*. Munich, Germany, 2003.
  92. Tang T.Y., McCalla, G. The Pedagogical Value of Papers: a Collaborative-Filtering based

- Paper Recommender. *Journal of Digital Information*, 10(2), 2009.
93. Tewari, G., Youll, J., Maes, P. Personalized location-based brokering using an agent-based intermediary architecture. *Decision Support Systems*, 34(2):127–137, 2003.
  94. Thorndike, E.L. A Constant Error in Psychological Ratings. *Journal of Applied Psychology*, 4(1):25–29, 1920.
  95. Triantaphyllou, E. *Multi-Criteria Decision Making Methods: A Comparative Study*. Kluwer Academic Pub, 2000.
  96. Tsai, K.H., Chiu, T.K., Lee, M.C., Wang, T.I. A Learning Objects Recommendation Model based on the Preference and Ontological Approaches. In *Proc. of the 6th IEEE International Conference on Advanced Learning Technologies (ICALT'06)*, 2006.
  97. Vincke, P. *Multicriteria Decision-Aid*. New York: J. Wiley, 1992.
  98. Vuorikari, R., Manouselis, N., Duval, E. Using Metadata for Storing, Sharing, and Reusing Evaluations in Social Recommendation: the Case of Learning Resources. *Social Information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively*, in Go D.H. & Foo S. (Eds.), pages 87–107. Hershey, PA: Idea Group Publishing, 2008.
  99. Wang, P. Recommendation Based on Personal Preference. *Computational Web Intelligence: Intelligent Technology for Web Applications*, in Zhang Y., Kandel A., Lin T., and Yao Y. (Eds.), World Scientific Publishing Company, 2004.
  100. Yu, C.C. Designing a Web-Based Consumer Decision Support Systems for Tourism Services. In *Proc. of the 4th International Conference on Electronic Commerce*, pages 23–25. Hong Kong, 2002.
  101. Yu, C.C. Innovation, management and strategy: A web-based consumer-oriented intelligent decision support system for personalized e-service. In *Proc. of the 6th International Conference on Electronic Commerce*, pages 429–437, 2004.
  102. Zeleny, M. *Linear Multiobjective Programming*. New York: Springer, 1974.
  103. Zimmerman, J., Kurapati, K., Buczak, A.L., Schaffer, D., Martino, J., Gutta, S. TV Personalization System: Design of a TV Show Recommender Engine and Interface. *Personalized Digital Television: Targeting Programs to Individual Viewers*, in Ardissono L., Kobsa A., Maybury M. (Eds.), Human-Computer Interaction Series 6, 2004.

# 具有健壮性的协同推荐

Robin Burke、Michael P. O'Mahony 和 Neil J. Hurley

**摘要** 协同推荐系统容易受到恶意用户的攻击，使得推荐结果产生偏差并倾向于推荐（或不推荐）某一些物品。从 2002 年开始这一类研究就属于热门研究方向。研究者已经发现研究最广泛的基于内存的算法都有相当简单的典型攻击漏洞。本章将讨论这些发现和研究结论。尤其包括伪造用户的检测方法以及具有健壮性的推荐算法的实现方法。

## 25.1 简介

协同推荐系统需要基于用户的正常行为。通过“协同”字眼我们可以看到某种潜在的意义，即一定程度上来说用户是站在一边的，至少都是希望推荐系统能提供好的推荐服务，同时也推荐有用的信息给和他们相关的人。Herlocker 等[10]用“茶话会”做类比，以此来描述同事之间进行善意交流的场景是与推荐场景相似的。

然而，现今的状况显示，网络并不仅包含善意且较配合的用户。在和推荐系统交互的时候用户拥有各种各样的意图，有些时候某些意图会和系统拥有者或者大部分的系统用户相悖。这里引用一个例子，Google 搜索引擎发现自己已经被卷入了一场持久战争中，战争的对方一直寻求跟检索算法博弈的方法，以提高自己所拥有的网站的检索排名。

在搜索引擎领域中，攻击者的目的就是使某些网页“看起来”像是 Google 在考虑所有因素之后对检索的请求所给出的最佳答案。在推荐系统领域也是相同情况。攻击者的目的就是要将某件商品或者物品伪装成一件“合适”的推荐物，推荐给相应的用户。攻击者也可能通过攻击手段阻止某件合适的物品被推荐出来。如果推荐系统仅依赖于用户信息，那么攻击者就一定会通过注入伪造的用户信息的方式去促进推荐系统的结果朝着自己所需要的方向走。单个的用户信息不足以改变推荐结果，推荐系统倾向于不受单个数据的影响。攻击者为了左右系统的推荐结果，必须伪造大量的用户。站长试着使这种伪造数据的注入行为代价高昂，当然也需要权衡一下，不能阻止推荐系统正常的数据的输入。通过设计用户评分数据来操控推荐结果的方法在文献[24]中第一次提到。从那以后，研究者开始聚焦于攻击策略、攻击检测策略以及设计及健壮的推荐算法。

图 25.1 中展示了研究框架。研究表明有效的攻击是能花费较少的代价对系统产生足够的影响。这样我们就可以理解图中在不同攻击效果下的影响力曲线。在检测攻击的研究中主要是识别出伪造的用户组并且从系统数据库中删除它们。低效的攻击很难被检测出

---

Robin Burke, Center for Web Intelligence, School of Computer Science, Telecommunication and Information Systems, DePaul University, Chicago, Illinois, USA e-mail: rburke@cs.depaul.edu

Michael P. O'Mahony, CLARITY: Centre for Sensor Web Technologies, School of Computer Science and Informatics University College Dublin, Ireland e-mail: michael.p.omahony@ucd.ie.

Neil J. Hurley, School of Computer Science and Informatics, University College Dublin, Ireland e-mail: neil.hurley@ucd.ie

翻译：胡聪（胡户主） 审核：潘泉波，左其盛



来。就是因为它们的攻击效果不好，所以需要大量的伪造用户才能产生效果。针对某种物品的大量的用户评分数据流入系统中是容易被系统监控工具检测出来的。检测算法的研究者主要聚焦于有效攻击的算法。从而使得图中的可检测区域扩大，进一步限制攻击的范围。同时研究者已经设计了一些算法能够对攻击更健壮，这些攻击的影响力曲线比有效攻击较低。通过综合上述研究结果，研究者也发现，并不需要彻底去除攻击效果，只要控制攻击的效果达到一个临界点，使得攻击不划算即可。

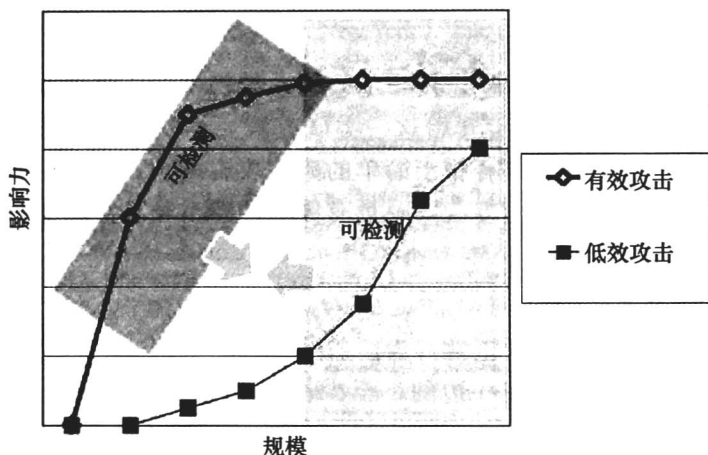


图 25.1 上图曲线对比了有效攻击和低效攻击的影响力。阴影区域表示容易被检测出来的攻击范围

本章将对要点逐一讲解。在 25.3 节，我们关注一些针对协同过滤推荐系统的攻击算法。25.5 节主要讲攻击的检测方法，尤其是图 25.1 中的左侧阴影区域所代表的有效攻击的检测方法。在最后 25.6 节将讲解如何通过健壮性的算法降低攻击的影响。

## 25.2 问题定义

协同推荐系统会根据用户新增的数据去调整推荐结果。我们可以稍微反直觉地假设，健壮性或稳定性是一种令人满意的系统自适应属性。攻击者会通过注入大量的用户信息来控制推荐结果，具有健壮性的推荐系统的目的就是要阻止这种攻击行为，即用户信息注入攻击。

我们假设任何的用户信息都是可用的，也就是说，我们并不需要用户的评分信息符合以前的评分，也不需要符合客观感受。用户可以随意选择，并且用户非常可能今天表现成这样，明天又会以一个新的身份登录并表现成另外一个样子。另外，一些类似攻击行为的网络现象并不是攻击。例如，在 2008 年秋天，大量的视频游戏粉丝聚集在亚马逊的 Spore 游戏主页上，大家只是在这个页面上抱怨游戏的软件版权问题。可以假定这些是大量的真实用户行为，并且他们的投票毫无疑问在一段时间内左右了 Spore 这款游戏的推荐效果。这些用户的行为不应该被当成攻击。目前还不清楚是否存在某种自动化的技术能够识别一个真实用户的评分行为是真实的还是开玩笑的，或是一次不诚实的行为。<sup>①</sup>

作为研究的目标，攻击是一种注入大量伪造的用户信息，用以达到影响推荐结果的

① 也许有人会争论是否存在这种方法，但是运用这种识别方法却不能给协同推荐系统带来什么好处。

目的的行为。攻击者所利用的每个独立的身份被称为攻击用户信息(attack profile)。这些攻击用户一旦被创建就被用来注入具有偏好的用户信息。最危险的攻击是那些对系统有重大影响的攻击,如此多的研究一直致力于在各种算法上寻求最有效和实际的攻击手段。

尽管会有随意的破坏行为,但这一领域的研究一直在关注一种专门为了达到某种特殊推荐效果的攻击。推举产品攻击和打压产品攻击分别是为了提高和降低某些被推荐物品的推荐值。此种攻击就是要降低竞争者的推荐效果,进而使得自己的物品能得到好的推荐效果。

对攻击者来说,好的攻击能够利用较少的努力去得到最大的影响力。有两类行为可以用于攻击。第一类是需要设计好被注入的用户信息。这种用户信息根据所需信息量又可以分为两种,充分信息攻击必须在攻击前知道推荐系统的详细评分分布。例如,一些攻击需要知道每个物品评分的均值和标准差。非充分信息攻击则只需要不依赖于具体系统的信息知识即可,像在公共信息源中能获取的知识。

我们通常假设攻击者能获取到推荐系统的算法类型。攻击者如果得到越多精确的推荐算法信息,则越能设计出有效的攻击,这种攻击会利用推荐算法的某种数学特性提高攻击效果。

第二类可用于攻击的事情是了解需要多少用户信息才会达到攻击效果。评分信息不够重要就是因为评分容易被自动化的攻击手段注入。许多网站会采用需要人工干预的在线注册方案,通过这种方法,站长就能要求每注册一个新用户都必须付出一定代价。这就是攻击者更倾向于那些需要注入更少用户信息的攻击手段的原因。

25.2.1 一个攻击示例

我们利用图 25.2 的简单推荐系统数据来演示基于用户信息注入的攻击原理。在这个例子中,攻击的目标是降低 7 号物品的推荐值(如产品打压攻击),为此有几个用户信息被注入(用户  $i$  到用户  $m$ )。

		物品						
		1	2	3	4	5	6	7
用户	a	+	-		+	+		+
	b	-	+	+	-	-		-
	c	+	-	+		-	-	-
	d	-	+	+	-			
	e	-		-	-	-		-
	f	+	-	+	+	+		+
	g		-	+	+	-	-	+
	h	+	-	+	+	+		?
	i	+	-	+		-	-	-
	j	-	+	+	-			-
	k	-		-	-	-		-
	l	+	-	+	+	+		-
	m		-	+	+	-	-	-

可信用户信息

目标用户信息

攻击用户信息

图 25.2 以上简单的系统数据展现了可信的用户信息以及一些被注入的攻击信息。在这个例子中,将预测用户  $h$  对 7 号物品的取舍,即产品打压攻击的目标

我们只需要考虑用户  $h$  是否喜欢 7 号物品。在第一个示例中, 我们忽略被注入的用户信息, 只考虑可信的用户信息(用户  $a$  到用户  $g$ )。不管用了哪种推荐算法, 推荐算法认为用户  $a$  到用户  $f$  的喜好代表现有的活跃用户, 并且这些用户都喜欢 7 号物品, 最终会使得推荐系统对 7 号物品做正向判断。

如果攻击信息也被考虑进去, 那么结果就毫无疑问会被改变。这些攻击信息的偏好也和用户  $h$  相似, 同时都不喜欢 7 号物品, 这样推荐系统就倾向于给予 7 号物品一个负面的评价。最终攻击的目标达到了。25.3 节将讨论被用来攻击的用户信息如何在真实的系统中产生作用。

## 25.3 攻击分类

基于伪造用户信息的攻击需要把许多用户信息单元注入推荐系统中。每个用户信息单元由一组(物品, 评分)单元组成。每个用户信息单元是一组物品的向量, 每个物品有一个评分值, 并且给予没有评分的物品 null 值。定义目标物品  $i_t$  作为攻击者的目标, 攻击者可能提高也可能降低该物品的推荐值。一般还会有随机挑选出一组可用的填充物品, 定义为物品集合  $I_F$ 。某些攻击模型也会利用数据库之外的物品集合。这组挑选的物品都和目标物品(或目标用户)有一定的相关性。对某些攻击来说, 这个集合是空的。可以定义为  $I_S$ 。最后, 定义集合  $I_\emptyset$  包含了那些没有被评分的物品。因为被预选出的物品集合较小, 每个用户信息的大小(评分总数)是由填充物品集  $I_F$  大小决定的。一些实验结果显示, 填充集的大小随着  $I$ (即所有物品的集合)的大小而定。

### 25.3.1 基础攻击

在文献[12]中最先描述了两基本的攻击模型, 随机攻击模型和平均攻击模型。这些攻击方法中都会包含伪造用户的构造方法, 攻击方法中利用随机评分的方法给填充物品评分。

#### 25.3.1.1 随机攻击

在随机攻击方法所利用的用户单元中, 对于填充物品的评分采取随机的评分值, 评分符合以所有物品评分均值为中心的数据分布。目标物品的评分值在推举攻击方法中被设置为最高评分值  $r_{\max}$ , 在打压攻击方法中被设置成最低评分值  $r_{\min}$ 。

发起攻击所需要的预备知识非常少, 因为所有系统的总体评分均值可以利用外部经验估出来(或者通过推荐系统本身得出)。然而随机攻击方法却并不十分有效[12, 6]。

#### 25.3.1.2 平均攻击

在文献[12]中描述了一种更有效的攻击方法, 利用每个物品的评分均值代替随机攻击方法中的全局均值(除了目标物品)。在平均攻击方法中, 每个填充物评分值的数据分布会(准确地或者相近地)符合该物品的评分平均值, 该平均值是由数据库中已有的用户评分计算得出的。

与随机攻击相同, 打压攻击利用  $r_{\min}$  评分代替推举攻击中的  $r_{\max}$ 。我们还应该注意的是, 平均攻击方法和随机攻击方法的唯一不同, 是用户单元中填充物的评分计算方法。

平均攻击方法可能需要有相当大的信息成本, 像需要知道填充物品列表的物品个数  $|I_F|$ 。这些填充物品的评分均值和标准差都需要提前知道。然而, 实验证明平均攻击方法即使在使用小规模填充物集合的情况下也能攻击成功。因此攻击所需的信息量也会大幅地

缩减,但是需要付出一些代价,伪造的用户都包含相同的物品列表,这样会使得这些伪造的用户引人注目[4]。

### 25.3.2 非充分信息攻击

平均攻击方法需要攻击者了解相当多的目标推荐系统的相关信息。一个合理有效的防卫方法就是尽量使攻击者难以获取系统的评分分布数据。下一个要描述的算法类型所需要的信息量将非常少。

#### 25.3.2.1 流行攻击

流行攻击的目的是把被攻击的物品和一小部分评分频率较高的物品关联起来。这个攻击利用了消费者市场的普及率的分布模型(Zipf 分布)。例如,畅销书这种物品在书籍整体中占比并不高,但是受到了绝大部分用户的关注和评分。攻击者用这个模型和关注率较高的物品去伪造用户信息。这种用户信息会更可能贴近大部分的用户,因为使用的物品都被大部分用户评分过。这个方法并不需要推荐系统相关的信息,因为很容易判定在任意的领域内哪个物品是最热门的。

流行攻击使用了被大部分用户所评过分的物品。这些物品和目标物品  $i_t$  都同时被赋予了最高分。填充物品的评分被随机选定,就像随机攻击方法一样。流行攻击因此也可以被看作随机攻击方法的一个拓展方法。

就像在 25.4 节中所展示的那样,对于基于用户的协同过滤算法<sup>①</sup>,流行攻击像平均攻击方法一样有效,但是又不用受“预先需要了解目标系统”的限制。因此这种攻击方法更容易发起。然而,就像平均攻击一样,攻击基于物品的协同过滤算法不够有效[12]。

#### 25.3.2.2 分段攻击

Mobasher 等在[19]中介绍了分段(segment)攻击方法,并证明了这种方法在攻击基于物品的协同过滤算法中的有效性。分段攻击的基本算法原理是,把某个物品引入一个目标用户群组中,同时这个目标群组的偏好已知或容易被预测出来。例如,恐怖电影的制片人可能希望自己的电影能被推荐给那些喜欢其他恐怖电影的观众。事实上,制片人其实并不想把他的电影推荐给不喜欢恐怖片的观众,因为会引起用户投诉,从而攻击行为也就被暴露了。

在发起此种攻击的时候,攻击者准备好一些目标用户群体所喜欢的物品集合。就像流行攻击一样,很容易知道某个用户群体所共同喜欢的物品。这些挑选出的物品和目标物品都被赋予最高评分值。为了最大化对基于物品的推荐算法的影响力,还可以对其他填充物品评最低分,进而最大化物品相似度的方差。

### 25.3.3 打压攻击模型

以上提到的所有攻击模型都可以用到打压攻击中。例如,就像之前提到的随机和平均攻击模型,把攻击目标的评分值与评分最小值  $r_{\min}$  联系起来。然而,25.4 节中的结果显示,相同的攻击算法对提升目标推荐效果有用,但不一定在压低目标推荐值方面能起到好的效果。因此,研究者额外又专门为打压攻击设计了一些方法。

① 参考第 4 章了解基于用户的协同过滤算法和基于物品的协同过滤算法的细节。

### 25.3.3.1 好恶攻击

好恶(love/hate)是一种简单的攻击方法。这种攻击并不需要对目标系统了解。在被用来攻击的用户信息单元中,目标物品被设置成最小评分值  $r_{\min}$ , 相反其他的填充物品的评分值设成最大  $r_{\max}$ 。这种攻击模型也可以被认为是热门(popular)攻击的非充分信息(low-knowledge)版本。令人吃惊的是,这个方法在攻击基于用户的推荐算法中效果最佳。

### 25.3.3.2 逆流攻击

逆流(reverse bandwagon)攻击是流行攻击的一个变种。攻击中选择出一些不怎么被用户评分的物品。这些物品和目标物品一起被设置成低分。因此目标物品就不被用户喜欢的物品关联起来了。进而推荐系统则会给目标物品较低的评价。这个攻击只需要事先了解较少的信息,如用户不喜欢的物品有哪些。在电影领域,这可能使电影在首映式前导致票房惨败。对于推荐系统方面的动作将高度优先于开幕式。

在 25.4 节中结果显示,针对压低目标推荐效果方面,尽管该方法在攻击基于用户的推荐算法时不如平均攻击有效,但是它却对基于物品的推荐算法十分有效。

### 25.3.4 知情攻击模型

非充分信息攻击有些近似平均攻击,会更多地关注热门的物品。攻击者最容易想到的攻击方法就是平均攻击,用户的比较是基于相似度的,进而伪造的用户会和处于平均范围的用户相近。如果攻击者能够获得更多目标系统的信息,那么也可以设计出更有效的攻击。

#### 25.3.4.1 热门攻击

假设推荐系统使用的算法是被广泛研究的基于用户的协同过滤算法[27]。用户相似度通过皮尔逊相似度计算<sup>①</sup>。像流行攻击一样,攻击用的用户信息单元会选出一组评分频率较高的物品。

伪造的用户信息和真实用户评分的物品如果相同也并不能保证相似性就高。流行攻击使用随机的填充物品评分来造成评分扰动,同时使得一部分用户信息能和某类固定的用户相关。热门攻击利用了物品的平均分,根据填充物品的评分是否高于平均来判断给予  $r_{\min} + 1$  还是  $r_{\min}$ 。将评分与物品的平均得分关联起来,容易使得伪造用户和真实用户的评分预测偏移值为正值(参考 25.4 节)。在文献[25]中详细描述了该算法。<sup>②</sup>

上面提及的用于推举攻击的评分策略可以稍加改动,然后应用于打压攻击。保持为正相关,同时评分预测偏移值为负,那么可以将策略改为,目标物品评为最低分  $r_{\min}$ , 依据被选择的物品的评分是否高于平均分,判断选定物品的分值是  $r_{\max}$  或者  $r_{\max} - 1$ 。

这个方法对于预备知识的需求量介于流行攻击和平均攻击之间。就像流行攻击,关注度高的物品很容易从系统中找到。因为没有填充物品,所以热门攻击将需要更多的热门物品。攻击者需要猜测物品的相对平均的偏好程度,作为评分的依据。这些评分信息可能从系统本身获取,如果不行,也可以从系统外部获取,如利用正向和负向的评论数来作为某种物品的被喜好倾向。

① 在文献[25]中讨论了知情攻击,其中提及多种相似性度量。并表示提到的相似性度量方法没有一种对攻击有抵御能力。

② 在文献[18]中也提到了一个最优推举攻击策略。这个示例的结论是最大化伪造用户与真实用户的相关性是最根本的目标。如果这个结论是对的,选择能最大化评分预测偏移值的伪造用户评分将很重要。



### 25.3.4.2 探测攻击策略

尽管没有针对热门攻击可检测性的研究,但这种攻击似乎很容易就能被检测出来,因为所有的攻击用户信息都完全相同,而且评分数据库中  $r_{\min} + 1$  和  $r_{\min}$  这样的评分也非常稀少。

一个略微效果差一点的方法是利用原本系统的推荐结果来获取合适的填充物品。此方法是探测(probe)攻击。攻击者只需要找到一个种子用户,并获取这个用户在推荐系统的推荐结果。这些推荐结果是通过种子用户的相邻用户产生的,所以至少能保证这些用户评分过这些推荐物品,并且预测的分数也能和这些用户相关。可以想象攻击者通过反复深入发掘可以影响一小组用户,或者扩大发掘范围则可以构建平均攻击。在某种意义上,探测攻击提供了一种针对推荐系统的增量学习的攻击方法。

相对于热门攻击,这种方法也有另外一种优点,即不需要太多推荐系统的预备信息。攻击者只需要选择少部分的种子物品,然后用推荐系统确定额外的物品和评分。如 25.4 节所示,种子物品被选出来并以类似于热门攻击的方法来评分。

## 25.4 检测系统健壮性

协同推荐算法可以分为两个类别,一般被称为基于内存的算法和基于模型的算法[2]。基于内存的算法利用系统中的所有可用数据来计算并预测哪些物品可以被推荐出来。相反,基于模型的算法是利用系统数据计算出一个模型,此后这个模型可以用在推荐的流程中。

研究文献中提出了大量协同推荐算法,关于这些推荐算法的健壮性研究并不是这章的主题。本章只针对两种主要推荐算法进行分析。它们是基于用户的推荐算法和基于物品的推荐算法[27, 32]。读者可以查阅文献[21, 20, 31]来获取其他协同推荐算法的健壮性分析。

### 25.4.1 评估矩阵

因为推举攻击和打压攻击的目标是提高或者压低目标物品的推荐值,我们需要评估它们的攻击效果。健壮性评估矩阵需要捕获目标物品的评分预测值和推荐状态(例如,是否被包含在 top  $N$  推荐列表)在攻击前和攻击后的区别。

许多研究者利用平均预测偏移值来评估评分预测值的变化量。定义  $U_T$  和  $I_T$  是测试用户集中的用户集合和物品集合。每个用户—物品对  $(u, i)$  的评分预测值的偏移量定义为  $\Delta_{u,i}$ , 可以利用公式  $\Delta_{u,i} = p'_{u,i} - p_{u,i}$  来计算,其中  $p$  和  $p'$  分别是攻击前和攻击后的评分预测值。如果偏移量是正值,则说明推举攻击成功地推荐了一个物品。物品  $i$  的平均预测偏移值的计算式为  $\Delta_i = \sum_{u \in U_T} \Delta_{u,i} / |U_T|$ 。所有物品的平均偏移值的计算式为  $\Delta = \sum_{i \in I_T} \Delta_i / |I_T|$ 。

如果强推某个物品的行为或者压制某个物品的行为可以表达攻击者的目的,那么预测偏移值是一个合适的指标。然而,有可能强推的物品的评分预测值已经有较大的偏差,但是还不足以使这个物品进入推荐列表。当该物品的初始评分值就很低时,这种情况就会发生,即使攻击使得评分值得到很大的提高也没用。为了计算攻击对推荐列表的影响程度,下面定义另一个矩阵:命中矩阵。定义  $R_u$  为用户  $u$  的前 top  $N$  推荐列表。如果目标物品出现在  $R_u$  中,则对于用户  $u$  来说  $H_{u,i}$  被定为 1, 否则为 0。物品  $i$  的命中率被利用以下

公式计算  $\text{HitRatio}_i = \sum_{u \in U_T} H_{ui} / |U_T|$ 。平均命中率利用所有物品中被攻击的物品  $i$  的命中率总和除以物品数量而得到： $\overline{\text{HitRatio}} = \sum_{i \in I_T} \text{HitRatio}_i / |I_T|$ 。

许多实验者利用公开的数据源 MovieLens 的 100K 大小的数据集。<sup>⊖</sup>这个数据集包含来自 943 个用户对 1682 部电影的 100000 个评分。评分值范围是 1~5，分值越高代表越喜欢。以后的实验结果都与这个数据集有关，除非有特殊说明。

#### 25.4.2 推举攻击

为了了解推举攻击的影响情况，我们可以查阅文献[20]中的描述。在这些图中，基于用户的推荐算法受到各种程度的攻击，这种攻击程度可以定义为注入的用户数在真实用户集中的占比。因此如果是 1% 的攻击程度，则对于 MovieLens 数据集来说需要注入 10 个用户。图 25.3 的左图中显示平均攻击使用了 3% 的填充物品，流行攻击用了一个评分次数最高的物品和 3% 的填充物品，随机攻击用了 6% 的填充物品。上面这些参数是该攻击的最优参数。毫无疑问，在预测偏差方面，预备知识量最密集的平均攻击的效果最佳。这种攻击非常有效。它能够把一个只有平均打分的电影(均值是 3.6)推荐到前五的位置。流行攻击尽管拥有最少的预备知识量需求，但是攻击效果却最接近平均攻击。此外，流行攻击显然优于随机攻击，更加重视有可能被许多用户评分的物品。

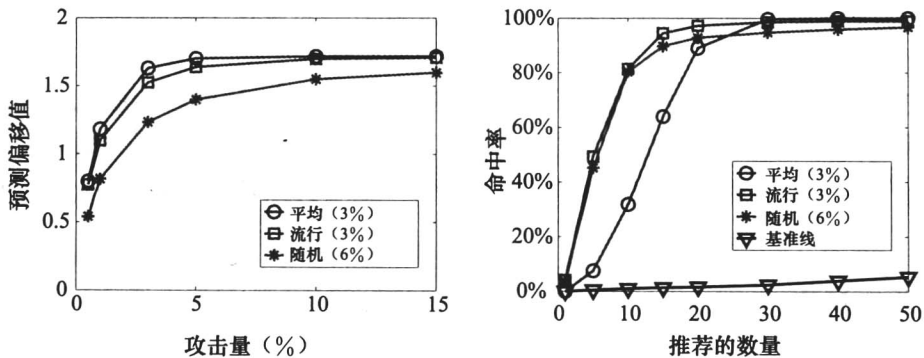


图 25.3 左图展现的是预测偏移值，右图是命中率。图中执行的是推举攻击，攻击目标是基于用户的协同推荐算法。命中率是在攻击量为 10% 时的图

有趣的是，图 25.3 的右图显示流行攻击拥有最大的攻击命中率。这意味着评分预测偏移值对于 top  $N$  推荐的评估来说并不一定是必要的。这个结果会促进攻击者利用流行攻击，因为它的预备知识量需求很少。需要注意的是所有攻击之后的命中率都明显地好于攻击前的命中率(图中定义的基线)。

文献[12]中描述了，基于物品的推荐算法对于平均攻击具有健壮性。文献[19]中描述，分段攻击是专门针对基于物品的推荐算法的一种低预备知识需求的攻击。它主要提升目标物品和用户所喜欢的物品的相似度。如果目标物品和用户喜欢的物品相似，则目标物品在系统中的评分预测值也会高——这是推举攻击的目标。所以攻击者的目标就是把自己的产品和流行的物品关联起来，让人觉着它们是相似的。喜欢这些相似物品的用户被认为属于一个目标群体。发起分段攻击的攻击者就是要选择出与目标物品相近的物

⊖ <http://www.cs.umn.edu/research/GroupLens/data/>

品作为伪造用户集  $I_s$ 。在电影的领域里，我们将会选择看流派相似的电影或者包含相同演员的电影。

文献[19]中，利用热门演员和电影流派来构建用户群组。如图 25.4 所示，选择的群组由以下用户组成：用户给 Alien、Psycho、The Shining、Jaws 和 The Birds 五部恐怖电影中至少三部的评分大于平均分(4 分或 5 分)。在这五部电影中，研究者列出所有三部电影的组合情况，其中每个组合至少有 50 个支持者，并且在从这些支持者中随机选择出 50 个用户并计算平均分。

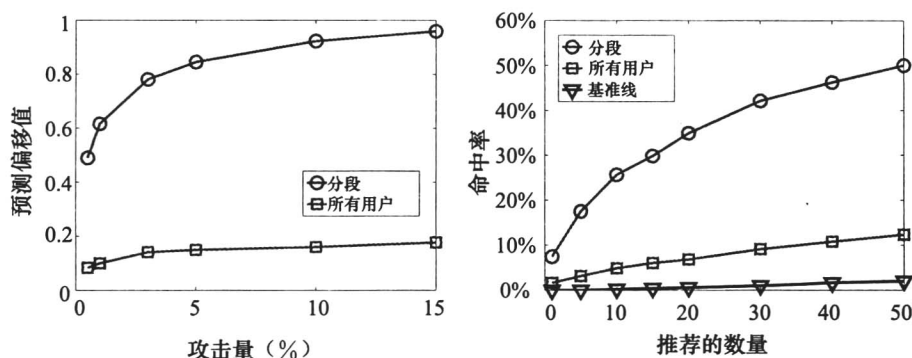


图 25.4 左图是评分预测偏移值，右图是命中率。实验是针对基于物品的协同过滤算法进行推荐攻击的。右图是在攻击量为 10% 的时候的命中率分布

在图中通过对比恐怖片粉丝群体和全部用户全体的情况，展示了分段攻击的效果。很明显分段攻击成功地把目标物品推荐到目标群体中。在攻击基于物品的推荐算法情况下，攻击效果可以和平均攻击比肩。例如，平均攻击在针对所有用户的前 10 推荐中能达到 30% 的命中率，此时攻击量为 10%。而分段攻击能获得相近的命中率，但是攻击量只需要 1%。

需要注意的是，尽管分段攻击是为基于物品的推荐算法而设计的，但是它对于基于用户的推荐算法也很好用。篇幅有限，我们不会在此展示实验结果，可以去文献[20]中获取详细资料。

### 25.4.3 打压攻击

可以假设打压攻击与推荐攻击相对称。它们唯一不同的是给目标物品的评分以及预测分值的影响力曲线。然而我们的实验发现一些有趣的现象，同样在一些攻击模型下推荐目标物品或压制目标物品，会在攻击效果上有差别。评分数据的分布情况应该作为考虑的因素，例如，在 MovieLens 数据集中低分数据很少，所以低分数据在目标物品的评分预测上占有较大影响。如在 top  $N$  推荐中，推荐出来的评分基线值(普通电影能进推荐列表的评分值)非常低，在大小为 50 的列表中基线值为 0.1。要把一个物品从推荐列表中压下去其实并不需要花费太多力气。

在好恶攻击中，随机选择的 3% 填充物品被赋予了最高分，目标物品被赋予最低分。对逆流攻击(针对基于物品的算法)来说，有着最低平均评分的物品在以系统的最低平均分为阈值选择了一些物品。在 25.3 节有详细的讨论。实验中的被选物品集合  $|I_s| = 25$ ，其中每个电影至少有 10 个用户评分。

在图 25.5 中展示了所有攻击模型的实验结果。尽管在好恶攻击中所需要的系统预备

知识很少，但是其在攻击基于用户的推荐算法中展示了最佳攻击效果。在其他打压攻击中，流行攻击超过了平均攻击，都与上文提到的推举攻击的结果不同。

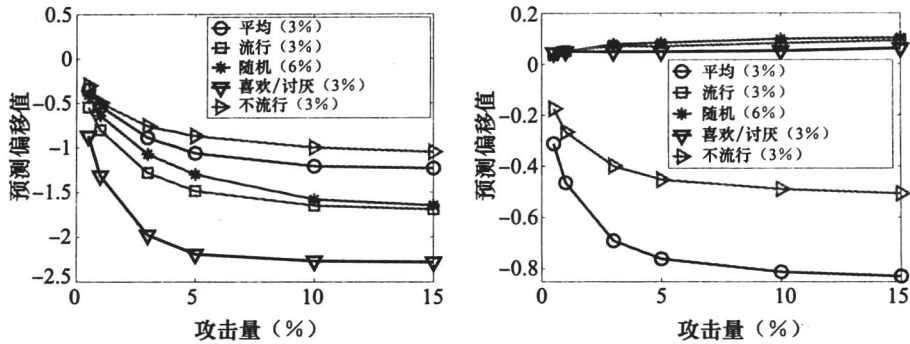


图 25.5 左图展示的是打压攻击对基于用户的推荐算法的评分预测偏移值，右图是针对基于物品的推荐算法

这些实验结果和推举攻击的数据的不相称之处令人惊讶。例如，好恶攻击在针对基于用户的算法的推举攻击中使用 10% 的攻击量，能够得到正向的评分预测偏移值并只略高于 1.0，这个攻击效果相对于随机攻击低不少。然而，当其用于打压基于用户算法下物品的时候，这个模型却是我们试过的最有效的模型。攻击的评分预测偏移值达到了平均攻击的两倍。在推举攻击中，平均攻击是最成功的，却在打压攻击方面是攻击效果最低的攻击之一。在推举攻击方面，流行攻击和平均攻击有相近的攻击效果，在打压攻击中也有较为优秀的攻击效果，尽管它只需要较少的系统预备知识。

总体来说，基于物品的推荐算法有更高的健壮性。针对基于物品的推荐算法，平均攻击是打压攻击中最有效的攻击，逆流行略低。推举攻击和打压攻击的不对称性在基于物品的算法攻击中得到验证。随机攻击和好恶攻击在推举攻击类别中属于差攻击效果的一类。但是在打压攻击类别中却得到了比较好的结果。在针对基于物品的推荐算法的打压攻击中，逆流行被证明是一种有效的、同时是低预备知识需求的攻击。

25.4.4 知情攻击

最后我们来评估一下针对基于用户推荐算法的知情攻击策略。特别地，我们会拿知情热门攻击与探测攻击与前面提到的平均攻击比较。

攻击实现如下所述。热门攻击的伪造用户信息包含 100 个物品(包括目标物品)，并且依照 25.3 节的规则进行评分。在探测攻击中，种子物品的选择方法如下，从推荐系统中选出 100 个最常被评分的物品，并随机从中选择 10 个。此后再不断获取通过系统给种子物品推荐出的物品和评分，补充伪造用户的物品列表。最终探测攻击需要 100 个物品。作为基准线的平均攻击也同样选择 100 物品，这将占到填充物品集的 1.7%。为了比较，平均攻击的伪造用户将使用 100 最常评分的物品(并不是像之前那样随机选择物品)。

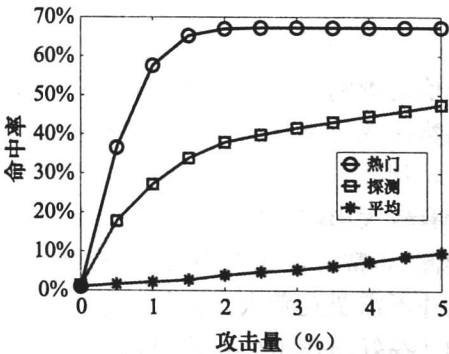


图 25.6 图中展示了热门、探测和平均攻击(推举攻击)对基于用户的推荐算法进行攻击的命中率曲线

图 25.6 展现了这三种攻击的命中率指数。图中显示知情攻击相对于平均攻击更加有

效。例如，当供给量为 2% 且取 top10 时，热门、探测、平均三个攻击的命中率分别为 65%、34% 和 3%。因此，考虑推荐算法的一些特殊属性被证明是有效的。

知情攻击的主要缺点是需要获取大量被攻击系统的信息才能选择出合适的物品和评分，进而构造出合适的伪造用户。像 25.3 节中所讨论的，这种攻击者用的预备信息是直接来自于作为攻击目标的推荐系统。这种预备信息可以从很多渠道获得，如最佳售卖者列表、物品的正向和负向的评论数等。即使有些场景下只能获取到部分信息，以前的实验结果也证明知情攻击也有很强的攻击效果[26]。

#### 25.4.5 攻击效果

通过如上研究总结，作为协同推荐系统主要部分的基于内存的算法很容易受到攻击。攻击者只需要掌握很少的系统信息就可以使得目标物品出现在用户的推荐列表中。这种有效的攻击是建造高稳定性和可用性的推荐系统的障碍。因此我们需要关注图 25.1 曲线中的规模低/影响力高的那一部分攻击。

为了解决这个问题，有两种对应的解决方案。一个是研究图 25.1 中的阴影检测区域，检测出伪造用户并删除。第二个方向是设计出受那些对经典算法奏效的攻击影响少的推荐算法。

#### 25.5 攻击检测

图 25.7 总结了攻击检测系统的组成部分。这是一个二元分类问题，对于每个用户信息有两种可能输出，也就是真实（即分类器把输入用户信息判定为真实的系统用户）和攻击（即分类器把这种输入用户信息判定为攻击者制造出来的）。像文献[7, 1]中所述的，一种检测攻击的方法是独立判定每个用户信息是否是一个攻击用户。见图 25.7 中的单个用户信息输入部分。这个输入是一个评分向量  $r_u$ ， $u$  为数据库中的用户。在进入分类器之前，会先通过特征提取步骤提取出一个特征集合， $r_u$  的特征集合是  $f_u = (f_1, \dots, f_k)$ 。 $f_u$  作为分类器的输入。输出是真实用户或者攻击用户。如果分类器是有监督分类器，它的训练集就是包含带标注的用户信息集合，例如，带有真实用户标签和攻击用户标签的用户单元集合。利用训练集可以学习出分类器的参数。

在许多攻击场景中，会通过许多组的攻击用户集攻击来达到推举或打压目标物品的目的。如文献[16, 23]中描述的那样，考虑攻击用户群组对构建有效的分类器很有帮助。在图中定义为“用户信息组”输入，分类器会把整个用户组当成一个输入。经过特征提取步骤之后，并给用户组中的每个用户标上标签。我们需要知道，在任意一个场景中并不是每个步骤都需要执行。例如，当  $f=r$  时，就可能没有特征抽取步骤；采用无监督分类器时，就没有必要经过训练阶段。

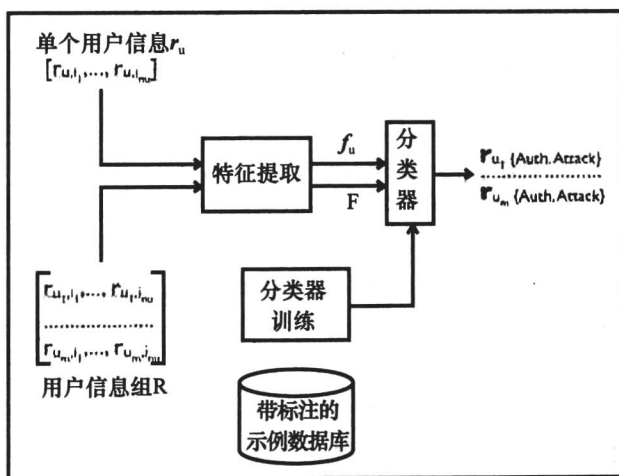


图 25.7 检测过程



### 25.5.1 评估矩阵

为了评估不同的检测系统，我们主要关心评估分类器的效果。“正向”分类表明一条用户信息被判定为攻击，分类数据中包含四种数据，如真正(true positive)和真负(true negative)代表被正确判定为攻击用户和真实用户。相反，另外两个是假正(false positive)和假负(false negative)，分别代表被错误地分类为攻击用户或者真实用户。文献中有许多评测方法计算以上定义的四种数据。遗憾的是，不同的研究者用了不同的评测方法，直接比较不同分类算法的效果往往很难。

精确度(precision)和召回率(recall)常用在信息检索领域中。在这种场景下，它们通过识别攻击来评估分类器的性能。每个评估量计算被正确分类的攻击用户的个数。召回率也被命名为灵敏度(sensitivity)，它代表被正确分类的攻击用户占有所有真正的攻击用户集合中的比例。精确度也可以被命名为正向预测值(positive predictive value, PPV)，代表被正确分类的攻击用户在所有被分类为攻击用户的集合中所占的比例。

$$\begin{aligned}\text{召回率} \equiv \text{灵敏度} &= \frac{\text{真正的数量}}{\text{真正的数量} + \text{假负的数量}} \\ \text{精确度} \equiv \text{PPV} &= \frac{\text{真正的数量}}{\text{真正的数量} + \text{假正的数量}}\end{aligned}\quad (25.1)$$

同样的评测方法也可以利用真实用户的识别率来计算。我们定义特异度(specificity)为被正确分类的真实用户的集合在所有真正的真实用户集合中所占的比例。同样也有负向预测值(Negative Predictive Value, NPV)，代表被正确分类的真实用户在所有被分类为真实用户的集合中所占的比例。

$$\begin{aligned}\text{特异度} &= \frac{\text{真负的数量}}{\text{真负的数量} + \text{假正的数量}} \\ \text{NPV} &= \frac{\text{真负的数量}}{\text{真负的数量} + \text{假负的数量}}\end{aligned}\quad (25.2)$$

在下面的实验结果中，我们会使用到精确度、召回率、特异度和 NPV。

#### 25.5.1.1 攻击性能

真实用户的误分类会导致真实的用户数据被删除，进而降低系统的推荐效果。一种评估这种影响的方法就是计算过滤攻击用户前后的推荐平均绝对误差(MAE)。当然从好的方面看，删除攻击用户会减少攻击影响。假设攻击是推举攻击或者打压攻击，评估攻击被影响的程度，可以计算攻击用户在检测过滤前后对目标物品的评分预测值。

#### 25.5.2 单用户检测

单用户检测的基础规则是攻击用户的评分分布会和真实用户的评分不同，因此每个攻击用户可以通过这些不同点被识别出来。就其本身而言，单用户检测是一种基于统计的检测问题。攻击者会不断地减小攻击用户和真实用户的统计差别，进而减小被检测出来的可能性。从另一个方面来说，一个划算的攻击会尽可能地包含特别有影响力的用户信息，例如，一个需要被推荐出去的物品应该被评为高分，而其他填充物品则需要被设置成能够推出目标物品的分值。结果，明显的攻击特征就体现出来了，以下是另外一些攻击特征：与系统的平均分差距太大，或者一条用户信息里面有不寻常的评分个数[1]。

### 25.5.2.1 无监督检测

在文献[7]中描述了一种无监督的单用户检测方法。检测基于攻击用户的普通属性。例如, 在用户信息里存在高于平常值的评分偏差, 这种用户单元会和相近的用户有超乎寻常的相似度。人们会衡量这些属性, 并用它们计算一条用户信息是攻击用户信息的概率。

### 25.5.2.2 有监督检测

基于监督的检测方法强调选择攻击用户所使用的特性, 并把这些特性提取成特征向量输入分类器。通常利用观察的方式, 会提取一些不同攻击方法都共有的特性, 也会从一些特殊的攻击算法中提取一些特殊属性。

在文献[5]中, 基于文献[7]中提出的哪些用户属性, 同其他类似属性一起被包含在一个特征向量里, 并输入有监督分类器。此外, 另外一些在用户信息中基于填充物和目标物品统计出来的特征属性(而不是整条用户信息)被提出来。例如, 填充物平均方差被定义为用户信息中被填充部分的评分的变化量, 这个特征被用来识别平均攻击。填充物平均目标差被定义为目标物品的平均得分和填充物品的平均得分, 被用来检测流行攻击。

作者使用了三种有监督的分类器: kNN、C4.5 和 SVM。kNN 分类器在训练集中, 利用用户信息的检测属性发现  $k=9$  的类别, 相似度使用的是皮尔逊相关系数(Pearson correlation)。C4.5 和 SVM 算法被以同样的方法构造, 只基于检测属性去分类用户信息。在图 25.8 中展现了针对 1% 攻击量的平均攻击的检测效果, 横轴是填充物品的数量。SVM 和 C4.5 在识别攻击用户方面具有完美的性能。但是它们相较于 kNN 算法错误分类了许多真实用户。在针对 1% 攻击量情况中, 在填充物品数量的整个范围内, SVM 的召回率和特异度是最佳的。

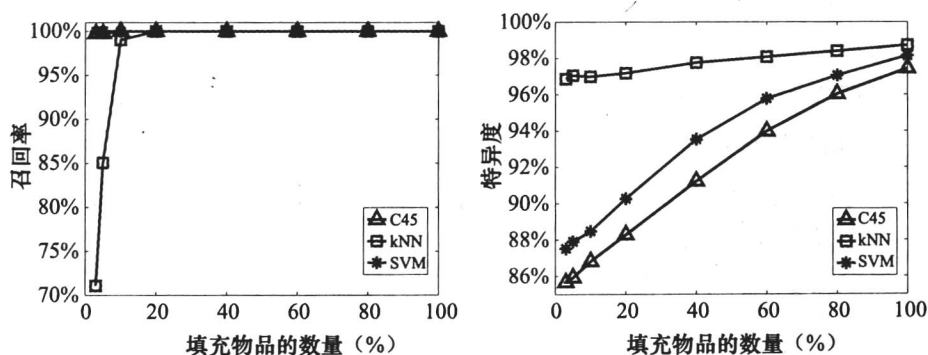


图 25.8 图中展现的是三种分类算法对 1% 攻击量的平均攻击的检测效果对比图, 左图是召回率, 右图是特异度

对真实用户的误分类所造成的影响程度通过计算过滤前后的 MAE 值来评估。通过观察, 当评分值为 1~5 的时候, MAE 值的增长会少于 0.05。在过滤攻击用户之后, 攻击有效性被大大地降低, 攻击的有效性是利用目标物品的评分预测偏移值来评估的。三种分类算法都很好地降低了攻击效果, 特别是在低攻击量的时候。SVM 算法在攻击量小于 10% 的时候, 在整个范围内能使评分不产生偏移。

### 25.5.3 用户组检测

在文献[23, 33, 18]中提出了一些无监督的伪造用户组识别方法。通常这些算法利用聚类算法来区分攻击用户组和真实用户组。

### 25.5.3.1 近邻过滤

文献[23]中提出了一个无监督检测和过滤方法。这个方法并不是在预处理阶段过滤用户信息，而是在给某个物品预测评分值的时候对活跃用户的近邻用户信息进行过滤。这个方法对于针对热门物品的攻击用户有优势。这个策略基于文献[8]中提出的算法。背景是信用评级系统，这个系统给在线市场中的买家和卖家进行评级，其中恶意机构会通过欺骗的行为恶意提高他们的信用评级。这个算法把近邻的用户组分为两个群。对类别中的数据进行分析，如果发现攻击用户，并找出在哪个组中包含攻击用户，在这个组中的所有用户信息将被删除。

聚类方法使用了文献[13]Macnaughton-Smith分裂聚类方法。然后，比较每个组内的热门物品的评分数据分布。因为攻击者的目标是把目标物品的评分值改成一个特殊值，所以有理由相信在攻击用户组中的目标物品的评分值会和真实用户组中的评分均值有较大差异。当两个聚集类别的评分均值差异较大时，可以判定发生过攻击。具有较小均方误差可以被判定为攻击用户组。

图 25.9 中展示了在 Movielens 数据集中，进行知情打压攻击时这个算法的检测评估结果(包括精

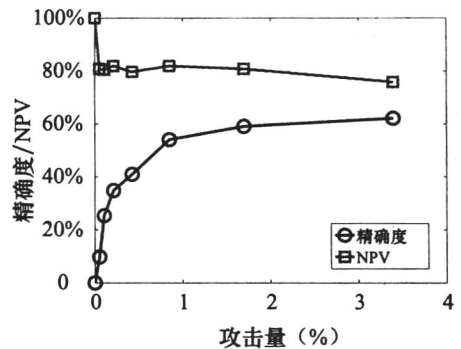


图 25.9 图中展示了近邻过滤方法的统计曲线。Y 轴是精确度和 NPV，X 轴是攻击量

确度和 NPV)。在所有攻击量下，真实用户被正确分类为真实用户的比率基本上在 75% 以上。可以说攻击用户被很好地过滤了。然而当攻击量比较小时，许多攻击用户被误认为真实用户。去除攻击用户的代价就是要去除一些真实用户，可能会影响评分预测的准确度。当去过滤一个没有被攻击的系统时，MAE 大概有 10% 的增长。

### 25.5.3.2 基于用户信息聚类检测攻击

在文献[18]中得到的观察是，攻击用户信息之间都互相高度相关，同时和许多真实用户很相似。这个发现在概率潜在语义分析(PLSA)和主成分分析(PCA)的帮助下，推动了采用聚类方法检测攻击方法的发展。

在 PLSA 模型中[11]，未观测因素的变量  $Z = \{z_1, \dots, z_k\}$  与每个观测值相关联。在协同过滤领域内，观测值相对应于用户—物品对的评分。并且用以下公式进行计算：

$$\Pr(u, i) = \sum_{i=1}^k \Pr(z_i) \Pr(u|z_i) \Pr(i|z_i)$$

利用期望最大值算法选择这个表达式的参数，来最大化观测数据的似然概率。像文献[21]中所讨论的那样，参数  $\Pr(u|z_i)$  被用来产生一类用户，如果  $\Pr(u|z_i)$  超过了阈值  $\mu$  将用户  $u$  定为类别  $C_i$ ，或者当  $\Pr(u|z_i)$  不超过阈值  $\mu$  的时候，最大化  $\Pr(u|z_i)$  也能定为该类。

在文献[18]中提到，大部分的攻击用户都倾向于被定为单独的一个用户组。识别包含攻击用户的用户组提供了一种有效的过滤方法。我们可以直觉地认为包含伪造用户的组内用户信息之间会更相似。每个组内的用户信息之间计算了马氏距离，并计算平均值，最后通过最小距离来过滤。实验证明利用 PLSA 来进行攻击检测，在强的攻击中效果很好，但是在弱的攻击中容易把攻击用户分散在不同的组中，导致检测效果差。

第二种方法是发掘攻击用户之间的高相似性。在文献[18]中描述了一种基于用户信息的协方差矩阵的 PCA 的检测方法。这种方法的本质就是将每个用户信息之间的方差最大化。PCA 被广泛用于高维数据的降维处理中。这种方法通过维度来识别用户类别,像文献[18]中描述的那样,通过 PCA 可以去除那些高度相关的维度。对于组  $C$ ,我们可以定义一个指示向量  $y$ ,其中当  $u_i \in C$  时  $y(i)=1$ ,否则  $y(i)=0$ 。定义  $S$  为协方差矩阵, $C$  中所有的用户信息的协方差和可以写成如下二项式形式:

$$y^T S y = \sum_{i \in C, j \in C} S(i, j)$$

$S$  的特征向量为  $x_i$ , 相关的特征值为  $\lambda_i$ , 其中  $\lambda_1 \leq \dots \leq \lambda_m$ , 二项式:

$$y^T S y = \sum_{i=1}^m (y, x_i)^2 (x_i^T S x_i) = \sum_{i=1}^m (y, x_i)^2 \lambda_i$$

文献[18]中所描述的方法可以理解成寻找最大化以上二项式的二元向量  $y$ , 其中比较小特征值所对应的 3~5 个特征向量具有弱相关性, 而大的特征值所对应的特征向量具有强相关性。

在图 25.10 中展现了 PLSA 和 PCA 的精确度和召回率, 实验的平均攻击量是 10%, 对于随机和流行攻击都有相似的结果。PLSA 和 PCA 算法需要指定过滤的范围, 在实验结果中, 过滤器大小根据攻击量来设定。所以需要参考来自于近邻过滤策略的结果(图 25.9), 而不需要去控制过滤器的范围。PLSA 获得最高 80% 的召回率, 是因为大概 20% 的错误分类。PCA 的处理效果不错, 即使攻击使用了随机、平均以及流行的混合形式的攻击用户。

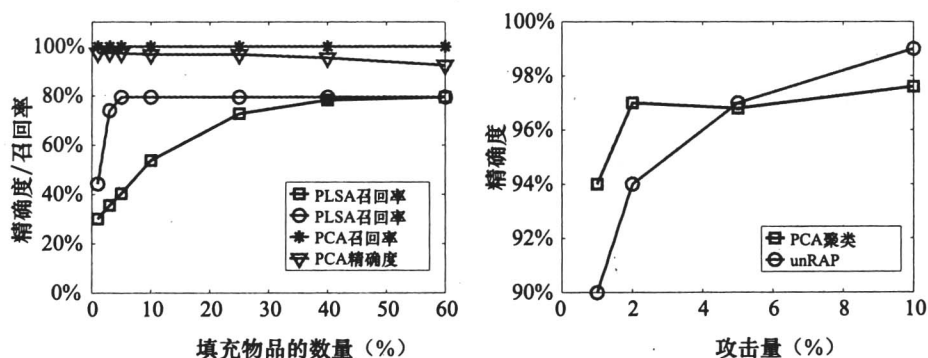


图 25.10 左图中展现了 PLSA 和 PCA 的精确度和召回率。其中 X 轴为过滤范围。针对的是 10% 攻击量的平均攻击。右图是 PCA 聚类方法和 UnRAP 的精确度, X 轴是攻击量, 攻击方法是平均攻击, 过滤范围为 10%

UnRAP 算法[3]用了聚类方法来区分攻击用户。这个算法用了一种测量值  $H_v$ , 这种方法被证明在识别高度相关的基因表达数据中非常有效。在攻击检测领域, 为每个用户计算  $H_v$ , 是用户评分均值的均方差, 物品均值的均方差, 和总体平均值。

$$H_v(u) = \frac{\sum_{i \in I} (r_{u,i} - \bar{r}_i - \bar{r}_u + \bar{r})^2}{\sum_{i \in I} (r_{u,i} - \bar{r}_u)^2}$$

$\bar{r}_i$  是所有用户对物品  $i$  的评分均值,  $\bar{r}_u$  是用户  $u$  对所有物品的评分均值,  $\bar{r}$  是所有用户对所有物品的评分均值。

数据库中的所有用户都计算了  $H_v$ , 并根据这个值做排序。前 10 个最高分值的用户被

识别为潜在的攻击用户，并从中检测出目标物品。目标物品是那种偏离用户评分均值的物品。下一步则将滑动窗口沿着用户排序列表向下滑动一个用户的位置。在窗口中计算目标物品的评分偏差的和，当这个和值达到 0 的时候停止。在这个迭代的过程中都存在的用户则成为候选的攻击用户。之后会在这个候选列表中过滤掉没有给目标物品评分或者和攻击用户相悖的用户。在图 25.10 中列出了这种方法在过滤平均攻击的情境中的精确度结果。并且和 PCA 组策略方法进行了比较。作者提到这种方法在攻击量中等的情况下效果好于其他方法，其他方法这种情况下效果会有所下降。

#### 25.5.4 检测结果

无论是有监督还是无监督的检测方法，在处理 25.3 节中的所有攻击方面都达到了理想的结果。这些结果是毫无疑问的，因为实验中的攻击都是精确设计过的，表现得相当规则，所以会和现实用户有差异。但是对现实中的攻击的处理效果却不得而知，因为一般网络公司都不愿意把自己的系统弱点给暴露出来。

回看图 25.1 中的框架图，以上的方法应该会对图中左上角的阴影处的可检测区域处理效果很好。对于善于攻击常用基于内存的算法的攻击方法也能被检测出来。现在依然遗留了一个问题，就是检测范围向下和向右发展进入那些不是最好的攻击方法的范围，但是这些攻击方法也可以为攻击者带来一定的利益。

### 25.6 健壮的推荐算法

相比于过滤和检测攻击，另一个方法是发掘对于攻击有健壮性的推荐算法。迄今为止，研究者已经提出了许多具有健壮性的推荐算法，这些算法主要是针对那些善于攻击基于内存的推荐算法的攻击。现有的一个未解决的问题是，新的攻击会专门挖掘这些健壮性好的推荐算法的弱点，这些推荐算法只是对于标准攻击有防范能力。

#### 25.6.1 基于模型的推荐

在文献[21]中描述了几种基于模型的推荐算法，这些算法针对那些高效攻击非常有效。而且这些算法并没有牺牲很多推荐准确率来提高健壮性。后续在文献[17, 15]中对基于模型的算法进行了综述，并提出了一种健壮的矩阵分解策略。

在文献[21]中提出了基于用户信息聚类的基于模型的推荐算法。在这个算法中，相同的用户信息被聚类为一组，之后再计算目标用户和用户组的相似性。对聚类后的每个组计算每个物品的评分均值，得到一个组级用户信息，之后利用组级用户信息来预测用户对其他物品的评分值。当用户  $u$  和一个用户组非常相似，那么可以利用这个组级用户信息对  $i$  的评分值来预测  $u$  对  $i$  评分。一般会选择  $k$  个最相似的组级用户单元的评分值来预测  $i$  的评分值，而不是用  $k$  近邻方法。在 25.5.3.2 节中描述了  $k$ -means 聚类方法和 PLSA-based 聚类方法，此处对这两种方法进行评估。对比  $k$ NN 算法，图 25.11 左图中展现了在平均攻击后的评分预测偏移值变化曲线。根据文献[21]中的描述，基于模型的算法被认为是相当健壮的算法，并且不会很明显地损失精确度。该文献中针对 PLSA 和  $k$ -means 聚类方法，用了 30 组级用户信息，计算 MAE 值分别为 0.75 和 0.76， $k$ NN 算法的 MAE 值是 0.74。



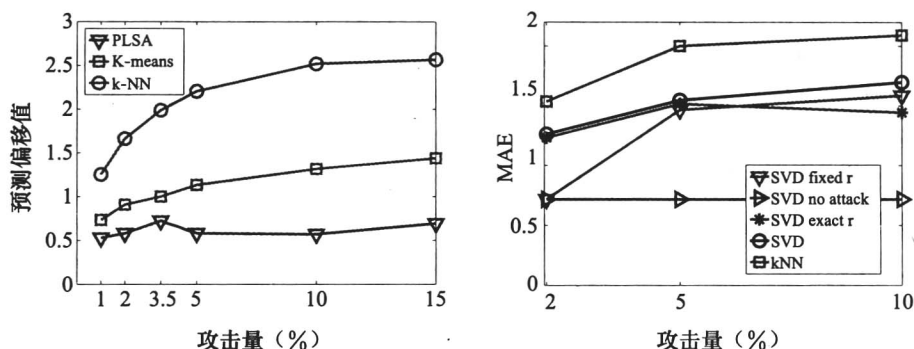


图 25.11 段推荐在平均攻击量为 5% 时预测变化和攻击量的比较(左图)。在攻击量为 10% 使用 RMF 情况下被攻击物品的 MAE 和攻击的变化比较(右图)

### 25.6.2 健壮的矩阵分解算法

有一种基于模型的推荐算法被证明非常有效,这种方法是一种矩阵分解算法,利用了奇异值分解方法(SVD)以及其变体。在文献[15, 18]提出了一种和聚类方法连用的矩阵分解方法,聚类方法在 25.5.3.2 节中提到了,并被用在了分解的训练阶段。像 PLSA 聚类方法和 PLSA 推荐方法可以联用。在文献[15]中提到,在过滤掉攻击用户组之后,留下来的用户组的  $\Pr(z_i|u)$  分布数据应该重新归一化,最后的几个训练步骤应该重新计算,进而能保留 PLSA 算法的评分预测精确度,并能显著减少评分预测偏移值。

在文献[18]中提到另外一个方法,利用了推广的赫布型学习算法(Generalized Hebbian Learning)来计算 rank-1 SVD 矩阵分解:

$$R \approx GH$$

$R$  是评分矩阵,  $G$  和  $H$  是 rank1 矩阵。这个算法被改进使得可疑用户对于评分预测模型的贡献值为 0。在图 25.11 的右图中展现了实验结果。图中展示了攻击算法的 MAE 值。其中可疑用户的数量和注入的攻击用户量设置成一样,量大概为所有用户的 7%。其中对比了 kNN 算法和标准 SVD 算法的 MAE 值,还有对比是否存在攻击用户的情况。

有一些支持基于模型的算法具有健壮性的理论结论。在文献[35]中提出了一种具有操控性的协同过滤算法并证明了这种方法的健壮性。方法中攻击对于单个终端用户的评分值数据影响减少了,同时增加了用户评分过的物品个数。此处的算法效果是根据攻击后的评分失真程度来评估的。以上的算法属于基于线性概率的协同过滤方法。本质上,系统模型的输出是评分分布的概率质量函数(PMF)。在线性算法中,被攻击的推荐系统的 PMF 值是以下两种情况的加权求和,一种是单纯考虑真实用户,另一种是单纯考虑攻击用户。当更多真实用户提供了评分数据,基于真实用户的 PMF 值就在整体的 PMF 值中占主要作用,健壮性就得到了保证。文中作者描述到,尽管在这种情况下最近邻算法不是线性的,但一些知名的基于模型的算法是渐进线性的,如朴素贝叶斯算法。

### 25.6.3 其他具有健壮性的推荐算法

攻击用户如果没在真实用户的近邻那就不能起到攻击效果。避免利用近邻选择规则计算相似性,那么那些和真实用户很相似的攻击用户就不能起到攻击效果,推荐算法的健壮性就能得到保证。文献[23]中描述近邻选择是去选择相近的并且最有用的用户信息来预测

评分。相似性只是可用性的其中一种评估方法,这种近邻规则可以利用其他评估方法来扩充。其中一种选择规则是反流行度。利用这种规则,推荐系统的推荐效果(利用 MAE 评估)能够得到保证。同时也能有效地压制那些基于热门物品的攻击用户所产生的攻击影响。

文献[31]中提出了一种具有健壮性的推荐算法,它基于关联规则挖掘算法。算法中把每个用户信息都当一个事务,并利用关联规则把相近的物品关联起来。定义物品集  $X \subset I$ , 用户信息集合中的一部分包含了这个物品集。关联规则定义为  $X \Rightarrow Y(\sigma_r, \alpha_r)$ , 其中  $\sigma_r$  是  $X \cup Y$ ,  $\alpha_r$  是规则的置信度, 定义为  $\sigma(X \cup Y) / \sigma(X)$ 。给用户  $u$  推荐一个物品会搜索基于关联规则的最高置信度的物品。其中  $X \subseteq P_u$  是用户信息子集,  $Y$  包含一些没有被  $u$  评分的物品  $i$ 。对于特定的物品没有足够的支持, 物品就不会出现在评分频繁的物品集中, 并且不会被推荐出来。这种算法对于平均攻击有效。在攻击量是 15% 的时候, 只有 0.1% 的用户被推荐过攻击物品。以上效果是基于关联规则算法。但是 kNN 算法却有 80% ~ 100% 的用户被推荐了攻击物品。但是作为交换, 关联规则算法的覆盖率低于 kNN 算法。并且这个算法对于分段攻击并不是健壮的。

#### 25.6.4 影响力限制器和基于信誉的推荐

在文献[28, 19]中提出了一种推荐算法, 其中健壮性的边界可以被计算出来。算法引入了两种额外的特性, 一种是影响力限制器(influence limiter), 一种是信誉系统。算法的思想是为每个用户设定一个权重, 即一种基于全局的信誉值。如果一个用户为一个近邻用户提供了正确的评分, 那么这个用户的信誉值提升, 否则下降。在这个推荐模型中, 作者提供了一些真实结果, 对于任何达到  $n$  个攻击用户的攻击策略, 攻击者所带来的负面影响都被一定程度限制了。某个用户如果想最大化影响力都会受限于信誉系统。这个算法的其他属性, 如精确度, 还在研究中。

影响力限制器是一种利用信誉度的推荐算法(见第 20 章)。近些年, 不断有将信任模型加入推荐的研究出现[14, 22, 9]。在文献[14]中信誉值用来在保持现有精确度的前提下增加推荐系统的覆盖率。在文献[22]中作者描述: 推荐系统中应该考虑带来精确推荐的用户信息。文献[9]中提出了另外一个算法中利用信誉值来过滤用户信息, 只是用信誉值高的前  $k$  个用户信息来预测评分。信誉值是基于用户对一个物品的是否正确评分来计算的。这个算法的健壮性对于攻击者获取信誉值的困难程度。

### 25.7 总结

协同推荐系统是可调节的, 因为当用户将他们的行为数据加入系统中时, 系统给出的推荐结果也会相应改变。在面对异常情况和错误情况时, 这个领域内的健壮性有别于一般经典计算机科学。我们的目标是保持系统可调节, 而不是让攻击者无可攻击。攻击者希望能够左右具有健壮性的系统输出, 那么他需要使得他的系统足够精巧, 不能让攻击检测系统检测出来, 设计的用户信息不会在新注册的用户群体中显得异常, 并且需要和现有真实用户的分布模型相匹配, 以至于降维方法不会将它们分离出来。如果以上这些都难以达到, 攻击的收益被限制了, 那么攻击者就找不到与他的努力相对应的攻击效果。这是在对抗性领域内所期望达到的效果。

很难说现在我们离这个期望的目标多近。攻击者如果发现某种检测策略实行了, 那么也会改进攻击算法来避免被检测出来。例如, 文献[23]中提到, 如果攻击者注意到某个规

则被用来决定用户的近邻内有没有攻击用户,攻击者会把攻击用户调节到不那么有效,使得它不被检测出来。在文献[34]中研究了不同攻击用户的模糊性评估方法。一般的研究发现,模糊的攻击用户信息相对于最优的攻击用户单元所带来的攻击效果并不差,并且更加难检测。这个方向需要更多的研究。

在对抗攻击的系统算法中提出了几个相近的方法。因为基于模型的算法比基于内存的算法有更高的健壮性,将会有针对基于模型的新的攻击方法出现。文献[31]中提出了针对分段攻击的关联规则方法。

我们可以从另外的角度来看这个问题,如推荐系统的设计者和攻击者是处在相同的一个游戏中。设计者设计了推荐系统,攻击者设计出最佳的攻击算法,然后设计者再给出回应修正系统,循环往复。我们想看到的结果是攻击者的收益越来越低,每一次迭代都使攻击需要更多的代价并且效果更差。带有检测策略的系统至少会比原始系统难以攻击,即使最差的情况下,即没有攻击被检测出来。至今还没有发现现有提出的健壮性推荐算法,如 RMF,具有明显的弱点,以至于容易受到一些精心设计的攻击。

## 致谢

Neil Hurley 感谢爱尔兰科学基金的支持,授权号 08/SRC/I1407; Clique: 图与网络分析聚类。Michael O'Mahony 由爱尔兰科学基金支持,授权号 07/CE/I1147; CLARITY: 传感器网络技术中心。

## 参考文献

1. A.Williams, C., Mobasher, B., Burke, R.: Defending recommender systems: detection of profile injection attacks. *Service Oriented Computing and Applications* pp. 157–170 (2007)
2. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence* pp. 43–52 (1998)
3. Bryan, K., O'Mahony, M., Cunningham, P.: Unsupervised retrieval of attack profiles in collaborative recommender systems. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 155–162. ACM, New York, NY, USA (2008). DOI <http://doi.acm.org/10.1145/1454008.1454034>
4. Burke, R., Mobasher, B., Bhaumik, R.: Limited knowledge shilling attacks in collaborative filtering systems. In *Proceedings of Workshop on Intelligent Techniques for Web Personalization (ITWP'05)* (2005)
5. Burke, R., Mobasher, B., Williams, C.: Classification features for attack detection in collaborative recommender systems. In: *Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining*, pp. 17–20 (2006)
6. Burke, R., Mobasher, B., Zabicki, R., Bhaumik, R.: Identifying attack models for secure recommendation. In: *Beyond Personalization: A Workshop on the Next Generation of Recommender Systems* (2005)
7. Chirita, P.A., Nejdl, W., Zamfir, C.: Preventing shilling attacks in online recommender systems. In *Proceedings of the ACM Workshop on Web Information and Data Management (WIDM'2005)* pp. 67–74 (2005)
8. Dellarocas, C.: Immunizing on-line reputation reporting systems against unfair ratings and discriminatory behavior. In *Proceedings of the 2nd ACM Conference on Electronic Commerce (EC'00)* pp. 150–157 (2000)
9. Fug-uo, Z., Sheng-hua, X.: Analysis of trust-based e-commerce recommender systems under recommendation attacks. In: *ISDPE '07: Proceedings of the The First International Symposium on Data, Privacy, and E-Commerce*, pp. 385–390. IEEE Computer Society, Washington, DC, USA (2007). DOI <http://dx.doi.org/10.1109/ISDPE.2007.55>
10. Herlocker, J., Konstan, J., Borchers, A., Riedl, J.: An algorithmic framework for performing

- collaborative filtering. In *Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval* pp. 230–237 (1999)
11. Hofmann, T.: Collaborative filtering via gaussian probabilistic latent semantic analysis. In: *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 259–266. ACM, New York, NY, USA (2003). DOI <http://doi.acm.org/10.1145/860435.860483>
12. Lam, S.K., Riedl, J.: Shilling recommender systems for fun and profit. In *Proceedings of the 13th International World Wide Web Conference* pp. 393–402 (2004)
13. Macnaughton-Smith, P., Williams, W.T., Dale, M., Mockett, L.: Dissimilarity analysis – a new technique of hierarchical sub-division. *Nature* **202**, 1034–1035 (1964)
14. Massa, P., Avesani, P.: Trust-aware recommender systems. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 17–24. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297235>
15. Mehta, B., Hofmann, T.: A survey of attack-resistant collaborative filtering algorithms. *Bulletin of the Technical Committee on Data Engineering* **31**(2), 14–22 (2008). URL <http://sites.computer.org/debull/A08June/mehta.pdf>
16. Mehta, B., Hofmann, T., Fankhauser, P.: Lies and propaganda: Detecting spam users in collaborative filtering. In: *Proceedings of the 12th international conference on Intelligent user interfaces*, pp. 14–21 (2007)
17. Mehta, B., Hofmann, T., Nejdl, W.: Robust collaborative filtering. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 49–56. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297240>
18. Mehta, B., Nejdl, W.: Unsupervised strategies for shilling detection and robust collaborative filtering. *User Modeling and User-Adapted Interaction* **19**(1–2), 65–97 (2009). DOI <http://dx.doi.org/10.1007/s11257-008-9050-4>
19. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Effective attack models for shilling item-based collaborative filtering system. In *Proceedings of the 2005 WebKDD Workshop (KDD'2005)* (2005)
20. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology* **7**(4) (2007)
21. Mobasher, B., Burke, R.D., Sandvig, J.J.: Model-based collaborative filtering as a defense against profile injection attacks. In: *AAAI*. AAAI Press (2006)
22. O'Donovan, J., Smyth, B.: Is trust robust?: an analysis of trust-based recommendation. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, pp. 101–108. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1111449.1111476>
23. O'Mahony, M.P., Hurley, N.J., Silvestre, G.C.M.: An evaluation of neighbourhood formation on the performance of collaborative filtering. *Artificial Intelligence Review* **21**(1), 215–228 (2004)
24. O'Mahony, M.P., Hurley, N.J., Silvestre, G.C.M.: Promoting recommendations: An attack on collaborative filtering. In: A. Hameurlain, R. Cicchetti, R. Traummüller (eds.) *DEXA, Lecture Notes in Computer Science*, vol. 2453, pp. 494–503. Springer (2002)
25. O'Mahony, M.P., Hurley, N.J., Silvestre, G.C.M.: An evaluation of the performance of collaborative filtering. In *Proceedings of the 14th Irish International Conference on Artificial Intelligence and Cognitive Science (AICS'03)* pp. 164–168 (2003)
26. O'Mahony, M.P., Hurley, N.J., Silvestre, G.C.M.: Recommender systems: Attack types and strategies. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI-05)* pp. 334–339 (2005)
27. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., J. Riedl: Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'94)* pp. 175–186 (1994)
28. Resnick, P., Sami, R.: The influence limiter: provably manipulation-resistant recommender systems. In: *RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems*, pp. 25–32. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297236>
29. Resnick, P., Sami, R.: The information cost of manipulation-resistance in recommender systems. In: *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pp. 147–154. ACM, New York, NY, USA (2008). DOI <http://doi.acm.org/10.1145/1454008.1454033>
30. Rokach, L.: Mining manufacturing data using genetic algorithm-based feature set decomposition, *Int. J. Intelligent Systems Technologies and Applications*, **4**(1):57–78 (2008).

31. Sandvig, J.J., Mobasher, B., Burke, R.: Robustness of collaborative recommendation based on association rule mining. In: RecSys '07: Proceedings of the 2007 ACM conference on Recommender systems, pp. 105–112. ACM, New York, NY, USA (2007). DOI <http://doi.acm.org/10.1145/1297231.1297249>
32. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In Proceedings of the Tenth International World Wide Web Conference pp. 285–295 (2001)
33. Su, X.F., Zeng, H.J., Chen, Z.: Finding group shilling in recommendation system. In: WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web, pp. 960–961. ACM, New York, NY, USA (2005). DOI <http://doi.acm.org/10.1145/1062745.1062818>
34. Williams, C., Mobasher, B., Burke, R., Bhaumik, R., Sandvig, J.: Detection of obfuscated attacks in collaborative recommender systems. In Proceedings of the 17th European Conference on Artificial Intelligence (ECAI'06) (2006)
35. Yan, X., Roy, B.V.: Manipulation-resistant collaborative filtering systems. In: RecSys '09: Proceedings of the 2009 ACM conference on Recommender systems. ACM, New York, NY, USA (2009)



## 本书贡献者名单

Gediminas Adomavicius

Department of Information and Decision Sciences

Carlson School of Management, University of Minnesota, Minneapolis, MN 55455,  
USA

e-mail: gedas@umn.edu

Xavier Amatriain

Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain

e-mail: xar@tid.es

Riccardo Bambini

Fastweb, via Francesco Caracciolo 51, Milano, Italy

e-mail: riccardo.bambini@fastweb.it

Gleb Beliakov

School of Information Technology, Deakin University, 221 Burwood Hwy,

Burwood 3125, Australia,

e-mail: gleb@deakin.edu.au

Robert Bell

AT&T Labs – Research

e-mail: rbell@research.att.com

David Bonnefoy

Pearltrees,

e-mail: david.bonnefoy@pearltrees.com

Peter Briggs

CLARITY: Centre for Sensor Web Technologies, School of Computer Science &  
Informatics, University College Dublin, Ireland,

e-mail: Peter.Briggs@ucd.ie

Robin Burke

Center for Web Intelligence, School of Computer Science, Telecommunication and

Information Systems, DePaul University, Chicago, Illinois, USA

e-mail: rburke@cs.depaul.edu

Tomasa Calvo

Departamento de Ciencias de la Computación, Universidad de Alcalá

28871-Alcalá de Henares (Madrid), Spain.

e-mail: tomasa.calvo@uah.es

Li Chen

Human Computer Interaction Group, School of Computer and Communication  
Sciences,

Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne,  
Switzerland

e-mail: li.chen@epfl.ch

Martine De Cock

Institute of Technology, University of Washington Tacoma, 1900 Pacific Ave,  
Tacoma, WA, USA (on leave from Ghent University)

e-mail: mdecock@u.washington.edu

Chris Cornelis

Dept. of Applied Mathematics and Computer Science, Ghent University, Krijgslaan  
281 (S9), 9000 Gent, Belgium

e-mail: Patricia.Victor@ugent.be

Maurice Coyle

CLARITY: Centre for Sensor Web Technologies, School of Computer Science &  
Informatics, University College Dublin, Ireland,

e-mail: Maurice.Coyle@ucd.ie

Paolo Cremonesi

Politecnico di Milano, p.zza Leonardo da Vinci 32, Milano, Italy Neptuny, via  
Durando 10, Milano, Italy

e-mail: paolo.cremonesi@polimi.it

Christian Desrosiers

Department of Software Engineering and IT, École de Technologie Supérieure,  
Montreal, Canada

e-mail: christian.desrosiers@etsmtl.ca

Hendrik Drachsler

Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit  
Nederland

e-mail: hendrik.drachsler@ou.nl

Boi Faltings

Artificial Intelligence Laboratory, School of Computer and Communication  
Sciences

Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne,  
Switzerland

e-mail: boi.faltings@epfl.ch

Alexander Felfernig

Graz University of Technology

e-mail: alexander.felfernig@ist.tugraz.at

Gerhard Friedrich

University Klagenfurt

e-mail: gerhard.friedrich@uni-klu.ac.at

Marco de Gemmis

Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona,  
4, Bari (Italy)

e-mail: degemmis@di.uniba.it

Ulrike Gretzel

Texas A&M University, 2261 TAMU, College Station, TX, USA,

e-mail: ugretzel@tamu.edu

Patrick J.F. Groenen

Econometric Institute, Erasmus University Rotterdam, The Netherlands,

e-mail: groenen@ese.eur.nl

Asela Gunawardana

Microsoft Research, One Microsoft Way, Redmond, WA,  
e-mail: aselag@microsoft.com

Andreas Hotho

Knowledge & Data Engineering Group (KDE), University of Kassel, Wilhelmshö,  
her Allee 73, 34121 Kassel, Germany,  
e-mail: hotho@cs.uni-kassel.de

Hans Hummel

Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit  
Nederland  
e-mail: hans.hummel@ou.nl

Neil J. Hurley

School of Computer Science and Informatics, University College Dublin, Ireland  
e-mail: neil.hurley@ucd.ie

Robert Jäschke

Knowledge & Data Engineering Group (KDE), University of Kassel, Wilhelmshö  
her Allee 73, 34121 Kassel, Germany,  
e-mail: jaeschke@cs.uni-kassel.de

Alejandro Jaimes

Yahoo! Research, Av.Diagonal, 177, Barcelona 08018, Spain  
e-mail: ajaimes@yahoo-inc.com

Simon James

School of Information Technology, Deakin University, 221 Burwood Hwy,  
Burwood 3125, Australia,  
e-mail: sjames@deakin.edu.au

Dietmar Jannach

TU Dortmund  
e-mail: dietmar.jannach@tu-dortmund.de

Martijn Kagie

Econometric Institute, Erasmus University Rotterdam, The Netherlands,  
e-mail: martijn@kagie.net

Dain Kaplan

Tokyo Institute of Technology, Tokyo, Japan  
e-mail: dain@cl.cs.titech.ac.jp

George Karypis

Department of Computer Science & Engineering, University of Minnesota,  
Minneapolis, USA  
e-mail: karypis@cs.umn.edu

Rob Koper

Centre for Learning Sciences and Technologies (CELSTEC), Open Universiteit  
Nederland  
e-mail: rob.koper@ou.nl

Yehuda Koren

Yahoo! Research,  
e-mail: yehuda@yahoo-inc.com

YoungOk Kwon

Department of Information and Decision Sciences

Carlson School of Management, University of Minnesota, Minneapolis, MN 55455, USA

e-mail: kwonx052@umn.edu

Pasquale Lops

Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona, 4, Bari (Italy)

e-mail: lops@di.uniba.it

Nikos Manouselis

Greek Research and Technology Network (GRNET S.A.)

56 Messogeion Av., 115 27, Athens, Greece

e-mail: nikosm@grnet.gr

Leandro Balby Marinho

Information Systems and Machine Learning Lab (ISMLL), University of Hildesheim, Marienburger Platz 22, 31141 Hildesheim, Germany,

e-mail: marinho@ismll.uni-hildesheim.de

Judith Masthoff

University of Aberdeen, AB24 3UE Aberdeen UK,

e-mail: j.masthoff@abdn.ac.uk

Lorraine McGinty

UCD School of Computer Science and Informatics, University College Dublin, Dublin 4, Ireland.

e-mail: lorraine.mcginity@ucd.ie

Kevin Mercer

Loughborough University,

e-mail: K.C.Mercer@lboro.ac.uk

Alexandros Nanopoulos

Information Systems and Machine Learning Lab (ISMLL), University of Hildesheim, Marienburger Platz 22, 31141 Hildesheim, Germany,

e-mail: nanopoulos@ismll.uni-hildesheim.de

Michael P. O'Mahony

CLARITY: Centre for Sensor Web Technologies, School of Computer Science and Informatics, University College Dublin, Ireland

e-mail: michael.p.omahony@ucd.ie

Nuria Oliver

Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain

e-mail: nuriao@tid.es

Jérôme Picault

Alcatel-Lucent Bell Labs,

e-mail: jerome.picault@alcatel-lucent.com

Pearl Pu

Human Computer Interaction Group, School of Computer and Communication Sciences,

Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne, Switzerland

e-mail: pearl.pu, li.chen, jiyong.zhang@epfl.ch

Josep M. Pujol

Telefonica Research, Via Augusta, 122, Barcelona 08021, Spain

e-mail: jmps@tid.es

Maryam Ramezani

Center for Web Intelligence, College of Computing and Digital Media, 243 S.

Wabash Ave., DePaul University, Chicago, Illinois, USA

e-mail: mramezani@depaul.edu

James Reilly

Google Inc., 5 Cambridge Center, Cambridge, MA 02142, United States.

e-mail: jamesreilly@google.com

Myriam Ribière

Alcatel-Lucent Bell Labs,

e-mail: myriam.ribiere@alcatel-lucent.com

Francesco Ricci

Faculty of Computer Science, Free University of Bozen-Bolzano, Italy

e-mail: fricci@unibz.it

Lior Rokach

Department of Information Systems Engineering, Ben-Gurion University of the  
Negev, Israel

e-mail: liorrk@bgu.ac.il

Neil Rubens

University of Electro-Communications, Tokyo, Japan,

e-mail: rubens@hrstc.org

Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL), University of

Hildesheim, Marienburger Platz 22, 31141 Hildesheim, Germany,

e-mail: schmidt-thieme@ismll.uni-hildesheim.de

Giovanni Semeraro

Department of Computer Science, University of Bari "Aldo Moro", Via E. Orabona,  
4, Bari (Italy)

e-mail: semeraro@di.uniba.it

Guy Shani

Department of Information Systems Engineering, Ben-Gurion University of the  
Negev, Beer-Sheva, Israel

e-mail: shanigu@bgu.ac.il

Bracha Shapira

Department of Information Systems Engineering, Ben-Gurion University of the  
Negev, Israel

e-mail: bshapira@bgu.ac.il

Barry Smyth

CLARITY: Centre for Sensor Web Technologies, School of Computer Science &  
Informatics, University College Dublin, Ireland,

e-mail: Barry.Smyth@ucd.ie

Gerd Stumme

Knowledge & Data Engineering Group (KDE), University of Kassel, Wilhelmshö  
her Allee 73, 34121 Kassel, Germany,

e-mail: stumme@cs.uni-kassel.de



Masashi Sugiyama  
Tokyo Institute of Technology, Tokyo, Japan  
e-mail: sugi@cs.titech.ac.jp

Panagiotis Symeonidis  
Department of Informatics, Aristotle University, 54124 Thessaloniki, Greece,  
e-mail: symeon@csd.auth.gr

Nava Tintarev  
University of Aberdeen, Aberdeen, U.K,  
e-mail: n.tintarev@abdn.ac.uk

Roberto Turrin  
Politecnico di Milano, p.zza Leonardo da Vinci 32, Milano, Italy Neptuny, via  
Durando 10, Milano, Italy  
e-mail: roberto.turrin@polimi.it

Alexander Tuzhilin  
Department of Information, Operations and Management Sciences  
Stern School of Business, New York University  
e-mail: atuzhili@stern.nyu.edu

Paolo Viappiani  
Department of Computer Science, University of Toronto, 6 King's College Road,  
M5S3G4, Toronto, ON, CANADA  
e-mail: paolo.viappiani@gmail.com

Patricia Victor  
Dept. of Applied Mathematics and Computer Science, Ghent University, Krijgslaan  
281 (S9), 9000 Gent, Belgium  
e-mail: Chris.Cornelis@ugent.be

Riina Vuorikari  
European Schoolnet (EUN), 24, Rue Paul Emile Janson, 1050 Brussels, Belgium  
e-mail: riina.vuorikari@eun.org

Michiel van Wezel  
Econometric Institute, Erasmus University Rotterdam, The Netherlands,  
e-mail: mvanwezel@acm.org

Kyung-Hyan Yoo  
William Paterson University, Communication Department, 300 Pompton Road,  
Wayne, NJ, USA,  
e-mail: toinette75@gmail.com

Markus Zanker  
University Klagenfurt  
e-mail: markus.zanker@uni-klu.ac.at

Jiyong Zhang  
Human Computer Interaction Group, School of Computer and Communication  
Sciences,  
Swiss Federal Institute of Technology in Lausanne (EPFL), CH-1015, Lausanne,  
Switzerland  
e-mail: jiyong.zhang@epfl.ch

# 翻译团队名单

## 第 1 章

翻译：燕山大学-王孝先

审核：王二朋，承皓，严强

## 第 2 章

翻译：吉林大学-丁彬钊

审核：刘湘勇，胡聪(胡户主)，郑州大学-吴宾

## 第 3 章

翻译：吴涛，吉林大学-丁彬钊，徐世超

审核：胡聪(胡户主)，北京大学-王雪丽，郑州大学-吴宾

## 第 4 章

翻译：刘诚

审核：吉林大学-丁彬钊，张彤彤，郑州大学-吴宾

## 第 5 章

翻译：王二朋

审核：吴亮，吴涛，郑州大学-吴宾

## 第 6 章

翻译：张思，沈耀

审核：严强，吴亮，郑州大学-吴宾

## 第 7 章

翻译：张杰

审核：郑重，洪弘，肖铨武，郑州大学-吴宾

## 第 8 章

翻译：北京大学-王雪丽，北京邮电大学-刘树栋

审核：吉林大学-丁彬钊，李曙光，郑州大

学-吴宾

## 第 9 章

翻译：刘良良

审核：邓展成，张杰，严强

## 第 10 章

翻译：潘泉波，胡聪(胡户主)，北京邮电大学-刘树栋

审核：张祺，吉林大学-丁彬钊，李青，郑州大学-吴宾

## 第 11 章

翻译：周俊宇，吉林大学-丁彬钊

审核：郭晓，张彤彤，吴亮

## 第 12 章

翻译：谢冰莹，余亮，郑州大学-吴宾

审核：吉林大学-丁彬钊，北京大学-王雪丽

## 第 13 章

翻译：赵华飞，卢青峰，郑州大学-吴宾

审核：李雪，吉林大学-丁彬钊，胡聪(胡户主)

## 第 14 章

翻译：韶关学院-梁永霖老师，燕山大学-王孝先，郑州大学-吴宾

审核：刘良良，左其盛，谢冰莹，吉林大学-丁彬钊

## 第 15 章

翻译：刘子微，吉林大学-丁彬钊，李雪

审核：张彤彤，郑州大学-吴宾

## 第 16 章

翻译：何伟岗，胡聪(胡户主)

审核：张洋，左其盛

## 第 17 章

翻译：何佳倍，燕山大学-王孝先

审核：程文君，胡聪(胡户主)

## 第 18 章

翻译：黎强，何佳倍

审核：张杰，胡聪(胡户主)

## 第 19 章

翻译：邓展成，佚名

审核：吴涛，胡聪(胡户主)

## 第 20 章

翻译：重庆工商大学-张天永

审核：李曙光，胡聪(胡户主)，郑州大学-  
吴宾

## 第 21 章

翻译：李艳民，张杰，佚名

审核：北京大学-王雪丽，郑州大学-吴宾

## 第 22 章

翻译：北京大学-王雪丽，马占国

审核：韶关学院-梁永霖老师，吉林大学-  
丁彬钊

## 第 23 章

翻译：北京大学-王雪丽

审核：刘良良

## 第 24 章

翻译：张秀伟

审核：朱亮，李青，李曙光

## 第 25 章

翻译：胡聪(胡户主)

审核：潘泉波，左其盛

# 推荐阅读



中文版  
第6版

作者: Abraham Silberschatz 著  
中文翻译版: 978-7-111-37529-6, 99.00元  
本科教学版: 978-7-111-40085-1, 59.00元



中文版  
第3版

作者: Jiawei Han 等著  
中文版: 978-7-111-39140-1, 79.00元



中文版  
第3版

作者: Ian H. Witten 等著  
中文版: 978-7-111-45381-9, 79.00元



中文版  
第2版

作者: Randal E. Bryant 等著  
书号: 978-7-111-32133-0, 99.00元



中文版  
第4版

作者: David A. Patterson John L. Hennessy  
中文版: 978-7-111-35305-8, 99.00元



中文版  
第6版

作者: James F. Kurose 著  
书号: 978-7-111-45378-9, 79.00元



中文版  
第3版

作者: Thomas H. Cormen 等著  
书号: 978-7-111-40701-0, 128.00元



中文版  
第2版

作者: Brian W. Kernighan 等著  
书号: 978-7-111-12806-0, 30.00元



中文版  
第7版

作者: Roger S. Pressman 著  
书号: 978-7-111-33581-8, 79.00元

## 推荐阅读



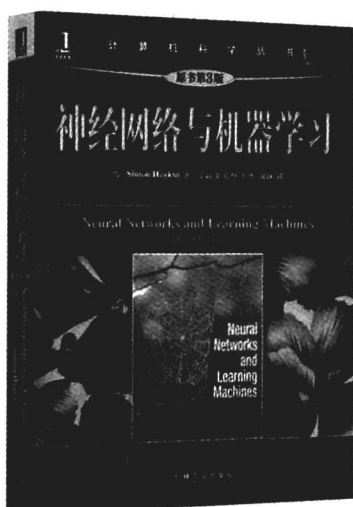
### 机器学习

作者：(美) Tom Mitchell ISBN: 978-7-111-10993-7 定价：35.00元



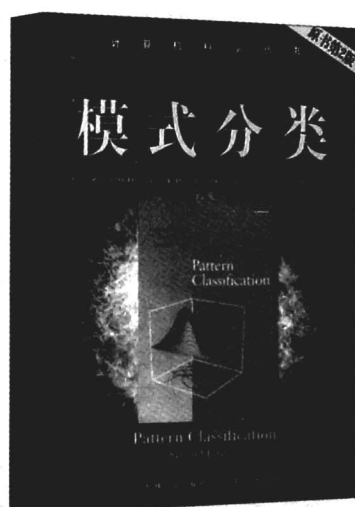
### 机器学习基础教程

作者：(英) Simon Rogers 等 ISBN: 978-7-111-40702-7 定价：45.00元



### 神经网络与机器学习 (原书第3版)

作者：(加) Simon Haykin ISBN: 978-7-111-32413-3 定价：79.00元

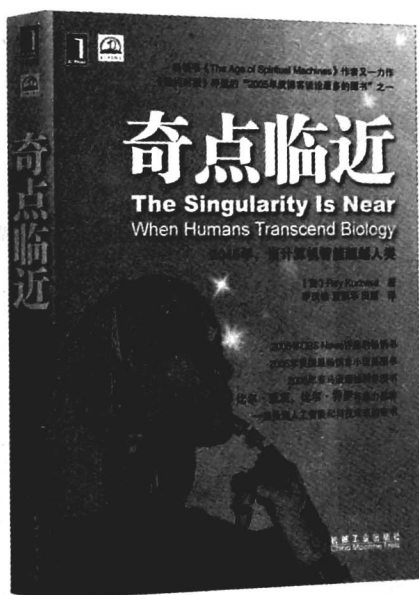


### 模式分类 (原书第2版)

作者：(美) Richard O. Duda 等 ISBN: 978-7-111-12148-1 定价：59.00元



## 推荐阅读



### 奇点临近

畅销书《The Age of Spiritual Machines》作者又一力作  
《纽约时报》评选的“2005年度博客谈论最多的图书”之一

2005年CBS News评选的畅销书

2005年美国最畅销非小说类图书

2005年亚马逊最佳科学图书

比尔·盖茨、比尔·乔伊等鼎力推荐

一部预测人工智能和科技未来的奇书

“阅读本书，你将惊叹于人类发展进程中下一个意义深远的飞跃，它从根本上改变了人类的生活、工作以及感知世界的方式。库兹韦尔的奇点是一个壮举，以不可思议的想象力和雄辩论述了即将发生的颠覆性事件，它将像电和计算机一样从根本上改变我们的观念。”

—— 迪安·卡门，物理学家

“本书对科技发展持乐观的态度，值得阅读并引人深思。对于那些像我这样对“承诺与风险的平衡”这一问题的看法与库兹韦尔不同的人来说，本书进一步明确了需要通过对话的方式来解决由于科技加速发展而引发的诸多问题。”

—— 比尔·乔伊，SUN公司创始人，前首席科学家